

HERRAMIENTA SOFTWARE PARA LA ENSEÑANZA DEL TEMA DE PRONÓSTICO

DANIEL SARMIENTO SUÁREZ

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA

2010

HERRAMIENTA SOFTWARE PARA LA ENSEÑANZA DEL TEMA DE PRONÓSTICO

DANIEL SARMIENTO SUAREZ

TRABAJO DE GRADO PARA OPTAR AL TÍTULO
DE INGENIERO DE SISTEMAS

DIRECTOR
INGENIERO ENRIQUE SARMIENTO MORENO

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA

2010

Tabla de contenido

INTRODUCCIÓN	11
1. MARCO CONTEXTUAL	13
1.1 OBJETIVOS	16
1.1.1 Objetivo General	16
1.1.2 Objetivos Específicos	17
1.2 VIABILIDAD	17
1.3 IMPACTO	18
2. MARCO DE REFERENCIA	20
2.1 PRONÓSTICO	20
2.2 SERIES DE TIEMPO	23
2.3 TÉCNICAS DE SUAVIZADO	25
2.4 MEDIA MÓVIL	31
2.5 MEDIA MÓVIL CENTRADA	32
2.6 SUAVIZADO EXPONENCIAL	34
2.7 DESCOMPOSICIÓN MULTIPLICATIVA	40
2.8 PROGRAMACIÓN ORIENTADA A OBJETOS	49
2.9 INTERFACES DE USUARIO	51
2.10 INTERFACES GRÁFICAS DE USUARIO	52
2.11 GUÍAS Y RECOMENDACIONES DE DISEÑO	53
2.11.1 Accesibilidad	53
2.11.2 Internacionalización y localización	54
2.11.3 Respuesta	54
2.11.4 Entrada del usuario	54
2.11.5 Otras recomendaciones	54
2.12 SISTEMAS OPERATIVOS	55
3. DISEÑO METODOLÓGICO	58
3.1 METODOLOGÍA DE DESARROLLO	58
3.2 FASES DEL DESARROLLO	60

3.2.1	Estudio Previo y Recopilación de Información	60
3.2.2	Planeación Proyecto y Definición de Metodología	60
3.2.3	Desarrollo	61
3.2.4	Documentación	62
3.2.5	Entrega y Soporte	62
4.	SOLUCIÓN Y RESULTADOS	63
4.1	HERRAMIENTAS DE DESARROLLO	63
4.1.1	Lenguaje de Programación: Python	63
4.1.2	Interfaz Gráfica: librería Qt	64
4.1.3	Cálculos Numéricos y Matrices: Librería NumPy	65
4.1.4	Gráficos Interactivos: Librería matplotlib	66
4.2	ORGANIZACIÓN DEL CÓDIGO	68
4.2.1	Módulo Pronóstico	68
4.2.2	Módulo ui_pronostico	71
4.2.3	Módulo estadística	71
4.2.4	Módulo mplwidget	72
4.3	FUNCIONAMIENTO	73
4.3.1	Cargar Datos	73
4.3.2	Cálculos	74
4.4	PRUEBAS	79
4.4.1	Prueba de Usabilidad	79
4.4.2	Prueba de Funcionamiento	82
4.4.3	Prueba Nuevo Material	84
5.	CONCLUSIONES	88
5.1	RESULTADOS	88
5.2	OBSERVACIONES	90
	BIBLIOGRAFÍA	92
	ANEXOS	93

Listado de Tablas

TABLA 2-1 TONELADAS DE MATERIA PRIMA POR ENTREGADAS POR PROVEEDOR EN LOS ÚLTIMOS AÑOS	26
TABLA 2-2 ERROR Y ERROR CUADRÁTICO	27
TABLA 2-3 DISTINTOS ESTIMADORES COMPARADOS	28
TABLA 2-4 VENTAS FABRICANTE PC 1985 - 1994	29
TABLA 2-5 MEDIA MÓVIL	31
TABLA 2-6 MEDIA MÓVIL CENTRADA	33
TABLA 2-7 TABLA FINAL DE MA CENTRADA	34
TABLA 2-8 PESOS PARA $\alpha = .3$	37
TABLA 2-9 VALORES DE α A MEDIDA QUE T AUMENTA	37
TABLA 2-10 ERROR PARA $\alpha = .1$	38
FIGURA 2-11 SERIE ORIGINAL Y SUAVIZADO EXPONENCIAL	39
TABLA 2-12 DATOS DEL EJEMPLO. 12.1	45

Listado de Figuras

FIGURA 2-1 CICLO DE VIDA DE UN PRODUCTO	21
FIGURA 2-2 SERIES DE TIEMPO CON TENDENCIA, ESTACIONALIDAD Y CICLICIDAD	25
FIGURA 2-3 VENTAS Y MEDIA	30
FIGURA 2-4 CICLO, INCONTROLADO VS. TRIMESTRE	48
FIGURA 2-5 ESTACIONALIDAD X INCONTROLADO ANUAL	48
FIGURA 2-6 ESTACIONALIDAD X INCONTROLADO SEMESTRAL	49
FIGURA 3-1: PROCESO SCRUM	59
FIGURA 4-1 QT DESIGNER	65
FIGURA 4-2 CÓDIGO COMPLETO PARA APROXIMACIÓN DE MÍNIMOS CUADRADOS.	66
FIGURA 4-3 WIDGET DE MATPLOTLIB INCRUSTADO EN LA HERRAMIENTA	67
FIGURA 4-4 APLICACIÓN CLÁSICA POR LOTES .VS. APLICACIÓN CON INTERFAZ GRÁFICA	69
FIGURA 4-5 OBJETO QSPINBOX PARA MODIFICAR EL NÚMERO DE PERIODOS EN EL AÑO	69
FIGURA 4-6 CONECTAR CAMBIO DE PERIODO CON MÉTODO QUE LO ATIENDA	70
FIGURA 4-7 MÉTODO CHANGE_ESTACIONALIDAD	70
FIGURA 4-8 FRAGMENTO MODULO UI_PRONOSTICO.PY	71
FIGURA 4-9 FRAGMENTO MÓDULO ESTADISTICA.PY	72
FIGURA 4-10 CLASE MPLCANVAS	72
FIGURA 4-11 VENTANA PRINCIPAL, DIALOGO ABRIR ARCHIVO	74
FIGURA 4-12 CÁLCULOS CON HERRAMIENTA SOFTWARE	75
FIGURA 4-13 CLASE MODELO MULTIPLICATIVO PARTE 1	76
FIGURA 4-14 CLASE MODELO MULTIPLICATIVO. PT 2.	78
FIGURA 4-15 PÁGINA PRINCIPAL CURSO ESTADÍSTICA GRUPO GEMA CON ENLACE A HERRAMIENTA SOFTWARE PARA PRONÓSTICO.	81
FIGURA 4-16 APLICACIÓN BAJO LINUX	83
FIGURA 4-17 EJECUTABLE WINDOWS XP (MÁQUINA VIRTUAL)	83
FIGURA 4-18 ARCHIVO BINARIO WINDOWS 7	84
FIGURA 4-19 DATOS VENTAS, TENDENCIA Y ESTACIONALIDAD	85
FIGURA 4-20 DATOS VISITAS SMASHITS.COM Y TENDENCIA (CUADRÁTICA)	86
FIGURA 4-21 PRONÓSTICO E INTERVALO DE 95% DE CONFIANZA. PUNTOS DE DATOS ACTUALES DEL AÑO PRONOSTICADO EN ROJO	87
FIGURA 5-1 CUADRO MODIFICAR GRÁFICO	89

RESUMEN

TÍTULO: HERRAMIENTA SOFTWARE PARA LA ENSEÑANZA DEL TEMA DE PRONÓSTICO.¹

AUTOR: DANIEL SARMIENTO SUAREZ²

PALABRAS CLAVE: pronóstico, series de tiempo, software educativo, estadística, promedio móvil, métodos de descomposición de series de tiempo, suavizado exponencial.

CONTENIDO:

El tema de pronóstico es de gran importancia en la práctica, ya que las predicciones de hechos futuros se suelen incorporar al proceso de toma de decisiones. Sin embargo, la enseñanza de este tema requiere de cálculos tediosos y repetitivos, que en ocasiones dificultan la comprensión de los conceptos y limitan la complejidad de los casos estudiados durante la asignatura.

Para apoyar el proceso de enseñanza, se desarrolló un software educativo de acompañamiento que proporciona al docente una herramienta para realizar los cálculos, procesar y visualizar el gran volumen de datos necesarios para aplicar modelos de pronóstico comúnmente utilizados en la práctica.

Con este software se busca aumentar el interés de los estudiantes por los temas de pronóstico y series de tiempo, apartándolos de cálculos y operaciones repetitivas que poco aportan al entendimiento de los mismos, y ampliando el número de problemas que se pueden mostrar como ejemplo. Actualmente, estos problemas de ejemplo están limitados a la cantidad de datos que se manejan en un medio impreso.

Debido al alto costo de software comercial para el manejo de pronóstico, se utilizaron tecnologías de software libre en el desarrollo de este proyecto. Este software fue desarrollado especialmente para de la asignatura Estadística II de la Escuela de Ingeniería de Sistemas de la Universidad Industrial de Santander.

¹ Proyecto de Grado

² Facultad de Ingenierías Físico Mecánicas. Escuela de Ing. de Sistemas. Director Enrique Sarmiento Moreno

ABSTRACT

TITLE: SOFTWARE TOOL TO AID IN THE TEACHING OF FORECAST.³

AUTOR: DANIEL SARMIENTO SUÁREZ⁴

KEYWORDS: forecast, time series, educational software, statistics, moving average, time series decomposition methods, exponential smoothing.

CONTENTS:

This graduation project aims to provide an educational software tool to address the need of a computer aid to support the teaching process of the forecast topic in the Statistics II Course. As this topic requires several repetitive calculations and large data sets handling that may impair the comprehension of key concepts.

Besides all the intermediate calculations required in order to arrive at the final answer, the vast amount of data necessary to apply commonly used forecasting models makes it necessary the aid of a computer to handle and visualize the data.

This educational software strives to solve the problems mentioned above, and is intended to increase the interest and comprehension of the forecast models and time series concepts taught in the course by allowing the instructor to use real world examples and eliminating the need to focus on calculations thus providing an interactive learning environment.

Due to the high cost of commercial software and the licensing restrictions in order to modify such software, open source libraries and programming languages were used in the development of this software tool.

This software was developed for the Statistics II course taught in the School of Systems Engineering at Universidad Industrial de Santander.

³ Graduation Project

⁴ Physical Mechanical Engineering Faculty. Systems Engineering and Informatics School. Director: Enrique Sarmiento Moreno.

INTRODUCCIÓN

Pronóstico es el proceso de realizar afirmaciones acerca de eventos que, generalmente, aún no han sido observados. El acto de realizar dichas afirmaciones se denomina pronosticar.

Pronosticar es muy importante en muchas y diversas áreas: pronósticos meteorológicos, pronósticos de tráfico en redes de telecomunicaciones, pronósticos del desempeño de equipos deportivos son algunos ejemplos. Existen cinco revistas A1 de sólo esta área.

Al predecir hechos que ocurrirán en el futuro, quien elabora los pronósticos debe confiar en la información de los hechos que han ocurrido en el pasado. La mayoría de las técnicas de pronóstico utiliza la siguiente estrategia elemental⁵:

- Analizar los datos, generalmente en una **serie de tiempo**; es decir, una sucesión cronológica de observaciones de una variable particular, para encontrar un patrón que se pueda utilizar para describirlos;
- Extrapolar o ampliar hacia el futuro el patrón encontrado con el objeto de preparar un pronóstico.

En este trabajo se presenta una herramienta que apoya los dos procesos anteriores y adicionalmente permite la medición de los errores de pronóstico.

⁵ Ver (Bowerman, O'Connel, Koehler; 2007)

El programa permite manipular los componentes de la serie de tiempo para estimar los parámetros que describen el comportamiento de la serie. Con estos parámetros se representa gráficamente el modelo de aproximación a la serie de tiempo y, al cambiar el valor de alguno de ellos, se muestra de manera inmediata, la forma en que estos cambios afectan el modelo de pronóstico.

Asimismo, una vez identificados los parámetros que describen la serie de tiempo, el programa realiza los cálculos necesarios para realizar predicciones con base en modelos de pronóstico y muestra los indicadores de error para la posterior validación del modelo por parte del usuario de la herramienta.

De acuerdo con el objetivo del proyecto, la finalidad educativa del software es apoyar al estudiante en el entendimiento del modelo, la visualización de los pasos intermedios necesarios para la consecución del resultado final, y la validación de dicho resultado.

GUÍA DE LECTURA

MARCO CONTEXTUAL: En este capítulo se presentan los objetivos a alcanzar con el desarrollo de este proyecto. En las dos últimas secciones de este capítulo se analiza la viabilidad e impacto de la herramienta desarrollada en este proyecto.

MARCO DE REFERENCIA: El objetivo de este capítulo es presentar al lector los conceptos básicos necesarios para la comprensión de la herramienta desarrollada en el proyecto. En la sección final se presenta una breve historia de los sistemas operativos así como recomendaciones de diseño de interfaces gráficas.

DISEÑO METODOLÓGICO: Se presenta la metodología de desarrollo utilizada y sus diferentes fases. Esta metodología se deriva del marco de desarrollo *SCRUM*.

SOLUCIÓN Y RESULTADOS: En este capítulo se muestran los resultados obtenidos y se valida la consecución de los objetivos propuestos. Se presentan las pruebas realizadas a la herramienta.

CONCLUSIONES Y RECOMENDACIONES: Finalmente, se realiza un balance del desarrollo del proyecto identificando las metas alcanzadas, oportunidades de mejora y recomendaciones para maximizar el impacto del proyecto.

1. MARCO CONTEXTUAL

A principios del siglo XX se hizo posible el acceso a calculadoras de bolsillo que permitían realizar una gran cantidad de operaciones matemáticas así como graficar datos. En su momento, muchos se opusieron al uso de calculadoras en el aula de clase, principalmente debido a las siguientes creencias:

Las calculadoras realizan el trabajo por el estudiante. De hecho, el pensamiento matemático involucrado en realizar cálculos mecánicos es prácticamente nulo. Una comprensión real de las matemáticas se logra como resultado de la comprensión del problema a resolver: ¿Qué se está preguntando? ¿Cómo se plantea el problema? ¿Qué operaciones son apropiadas? ¿Tiene sentido la respuesta obtenida?

Ninguna de estas preguntas puede ser respondida por una calculadora, y rara vez un estudiante que no tenga una comprensión del tema de estudio puede responderlas con tan solo el resultado arrojado por la calculadora.

Debido a que las calculadoras realizan todo el trabajo, el estudiante no es estimulado o desafiado suficientemente. Las calculadoras solo realizan tareas de bajo nivel, cálculos; ellas no “piensan”

Las calculadoras le permiten a los estudiantes trabajar suficientes problemas para descubrir y observar patrones en las matemáticas, los cuales rara vez son observados con los métodos manuales. Los alumnos, que en el pasado prácticamente se limitaban a memorizar reglas y fórmulas, se pueden enfocar en aplicaciones útiles y prácticas de los conceptos y teorías aprendidos en clase.

A pesar de todos sus beneficios y capacidades, las calculadoras nunca podrán reemplazar la mente humana en comprender un problema, escribir una ecuación apropiada para el problema, elegir las operaciones para resolver el problema e interpretar correctamente la solución. La calculadora es sólo tan efectiva como el estudiante que digita la información.

La calculadora es el equivalente al procesador de texto para los estudiantes de literatura. Los procesadores de texto no “crean” ensayos pero sí facilitan considerablemente su creación,⁶ al igual que las calculadoras no “entienden” las matemáticas pero si facilitan su entendimiento.

⁶ Ver (Waits, 1994)

Hoy en día, dos décadas después, la calculadora es utilizada por la gran mayoría de los estudiantes de colegios y universidades y hay varios estudios que demuestran las ventajas de incorporar calculadoras a la enseñanza. Ventajas que van desde la disminución de la ansiedad, aumento del entusiasmo, mejor actitud hacia las matemáticas por parte de los estudiantes que utilizan calculadoras⁷ hasta una disminución de la brecha de rendimiento entre hombres y mujeres gracias al uso de calculadoras en el salón de clase.⁸

Teniendo en cuenta que las calculadoras no son más que un computador programado para un propósito específico, podemos obtener los mismos beneficios desarrollando una herramienta software que permita la mejora de habilidades analíticas y eficiencia en procesos de pensamiento complejos, en lugar del desarrollo de simples habilidades mecánicas de cómputo.

Una herramienta que permita estudiar problemas representando situaciones de la “vida real” con datos complejos. De esta manera los estudiantes no tendrán que preocuparse si van a cometer un error algebraico o aritmético que no les permita llegar a la respuesta correcta, así hayan trabajado el problema de manera adecuada.

El uso de herramientas gráficas extiende el entendimiento del problema haciendo práctico y posible el uso de un concepto importante en la teoría de la enseñanza llamado múltiples representaciones enlazadas. Los conceptos estadísticos estudiados en clase ahora podrán ser resueltos mediante representaciones

⁷ Ver (Ray & Dessar, 1986)

⁸ Ver (Dunham, 1995)

numéricas, gráficas y simbólicas⁹. El software educativo a desarrollar permite cambiar fácilmente de una representación a otra.

Cuando los estudiantes pueden escoger entre distintos métodos de solución de un problema, es más probable que recuerden cómo resolver el problema y sean capaces de resolver un problema similar la próxima vez que lo encuentren. Además, ver los datos de manera gráfica ayuda a reforzar conceptos abstractos y permitir que el problema sea más tangible. Los estudiantes pueden utilizar la herramienta software para confirmar una respuesta hallada de manera manual o para resolver problemas que sean muy difíciles de resolver sin la ayuda del computador.

1.1 OBJETIVOS

1.1.1 Objetivo General

Desarrollar un software educativo de acompañamiento en el proceso de enseñanza del tema de pronóstico en la asignatura de Estadística Inferencial.

La finalidad educativa del software es apoyar al estudiante en el entendimiento del modelo y metodología de pronóstico utilizada, la visualización de los pasos intermedios necesarios para la consecución del resultado final¹⁰, y la validación de dicho resultado

⁹ Ver (Quesada, 1996)

¹⁰ Ya sea un intervalo de pronóstico o la estimación de todos los parámetros del modelo

1.1.2 Objetivos Específicos

- Suministrar una herramienta de software educativo que mejore y agilice el aprendizaje de los estudiantes e intensifique la retroalimentación con el docente.
- Brindar una herramienta que sirva como ayuda al profesor para presentar material didáctico.
- Ampliar la complejidad y cantidad de material didáctico disponible para el estudio del tema de pronóstico.
- Facilitar una ayuda interactiva para el estudiante en el aprendizaje de temas que son mejor asimilados si se cuenta con recursos didácticos interactivos multimedia.

1.2 VIABILIDAD

En el presente proyecto se tienen en cuenta aspectos factibles que apoyan los siguientes niveles para el análisis, diseño y desarrollo de esta herramienta:

- Se contó con el respaldo del director de proyecto y el grupo GEMA¹¹ quienes aportaron su experiencia y conocimiento en el desarrollo de software educativo.

¹¹ Grupo de Investigación en Comunicación Educativa, Escuela de Ingeniería de Sistemas e Informática UIS.

- Se contó con un grupo de estudiantes, usuarios de prueba, que, al interactuar con la herramienta, ofrecieron retroalimentación invaluable para el desarrollo del software.
- Gracias al uso de herramientas de software libre no se incurrió en costos de licencias de software de desarrollo.
- Se contó con el conocimiento técnico y experiencia por parte del autor en el manejo de las herramientas de desarrollo seleccionadas.

Estos recursos humanos, técnicos, y económicos así como resultados exitosos con herramientas educativas similares en semestres, entre los cuales se pueden destacar más de decena de simuladores para las asignaturas Estadística II Estadística II, anteriores permitieron tener confianza en la viabilidad de este desarrollo.

1.3 IMPACTO

Con el desarrollo de éste proyecto se busca aumentar la capacidad de los estudiantes de la materia Estadística II de comprender la teoría del tema de pronóstico y aplicar esta teoría en la resolución de problemas, apoyados en herramientas de software. Al aumentar la complejidad de los problemas se espera que los estudiantes puedan aplicar estos conceptos en problemas que probablemente serán encontrados posteriormente en su desempeño profesional.

La herramienta también será de vital importancia a la hora de validar los modelos y permitirá al estudiante experimentar rápidamente al modificar los parámetros del modelo de pronóstico y visualizar de manera interactiva los cambios, así como verificar la validez del nuevo modelo ya sea analizando indicadores de error calculados o contrastando los valores pronosticados con los valores reales de la serie de tiempo.

El apoyo a la labor docente se puede sintetizar en: nuevos contenidos, nuevos instrumentos y recursos para la docencia y su gestión, acceso a nuevos tipos de información, nuevos escenarios educativos y nuevos métodos pedagógicos¹².

La Universidad, y en especial la Escuela de Ingeniería de Sistemas, contarán con una herramienta de software, abierta y bien documentada que podrá ser extendida posteriormente para trabajar con nuevos modelos de pronóstico o tratar temas relacionados como la auto correlación.

¹² Software educativo para la asignatura de estadística en los temas de inferencia estadística y prueba de hipótesis – Chimá, García UIS - Proyecto Grado 2005

2. MARCO DE REFERENCIA

2.1 PRONÓSTICO

Como se mencionó en la introducción, pronóstico es el proceso de realizar afirmaciones acerca de eventos que aún no han sido observados. El acto de realizar dichas afirmaciones se denomina pronosticar.

La mayoría de las técnicas de pronóstico utiliza la siguiente estrategia elemental:

- Analizar los datos para encontrar un patrón que se pueda utilizar para describirlos;
- Extrapolar o ampliar hacia el futuro el patrón encontrado con el objeto de preparar un pronóstico.

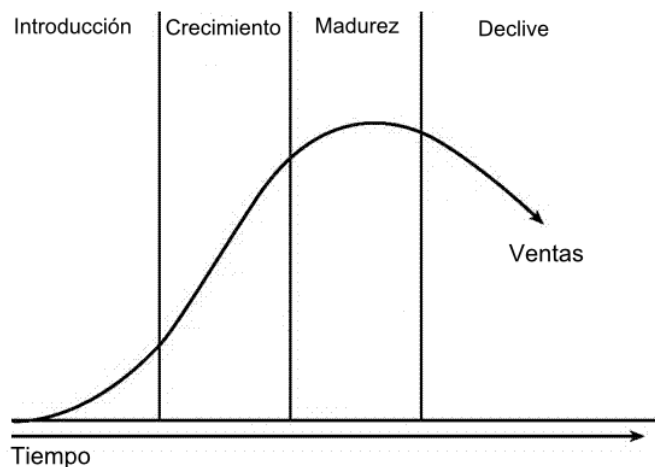
Existen diferentes métodos para establecer pronósticos los cuales pueden ser clasificados en dos tipos básicos: métodos cualitativos y cuantitativos.

Por lo general, en los **métodos cualitativos** se utiliza la opinión de un experto para predecir en forma subjetiva los hechos.

Se recurre a las técnicas cualitativas para establecer pronósticos cuando no se cuenta con datos existentes; por ejemplo, se podría considerar la situación en la que se introduce un nuevo producto al mercado. Para pronosticar las ventas de este producto, una compañía debe confiar en la opinión de expertos para realizar el pronóstico de las ventas de este nuevo producto.

Una técnica empleada puede ser ajustar una curva conocida de manera subjetiva. Por ejemplo se puede considerar la gráfica del ciclo de vida del producto la cual se presenta en la Figura 2-1. Si se desean pronosticar las ventas los expertos de la empresa podrían construir de manera subjetiva la curva de la fase de crecimiento, la cual se conoce como curva S.

Figura 2-1 Ciclo de vida de un producto



Para construir esta curva S, la empresa debe utilizar su conocimiento del mercado y experiencia con productos similares. La construcción subjetiva de esta curva es un proceso difícil que requiere gran experiencia y criterio.¹³

Los **métodos cuantitativos** para establecer pronósticos requieren el análisis de información anterior para establecer un modelo de pronóstico. Estos modelos para establecer pronósticos se pueden agrupar en dos clases: modelos univariantes y modelos causales.

¹³ Ver (Bowerman, O'Connell, Koehler; 2007)

Un **modelo univariable** predice los valores futuros de una serie de tiempo teniendo en cuenta sólo los valores anteriores de la misma serie de tiempo. Estos modelos son más útiles cuando se espera que los parámetros estimados para el modelo no cambien. Por ejemplo, un modelo univariable se puede aplicar para predecir el número de aspirantes a ingresar en una universidad, si se espera continuar usando la misma política de admisiones, pero este modelo no sería útil para predecir un cambio en el número de aspirantes después de realizar un cambio en las políticas de admisión, o un incremento en el puntaje mínimo del examen del ICFES para poder aspirar a ingresar a la universidad.

Los **modelos causales** para pronóstico describen la relación entre las variables que inciden sobre la variable que se desea pronosticar. Por ejemplo, las ventas mensuales de un producto podrían estar relacionadas con la temporada del año, el precio del producto y el índice de confianza de los consumidores.

El trabajo para establecer un modelo de pronóstico en este caso es del de estimar por medios estadísticos la relación entre las variables dependientes, en este caso las ventas, y las variables independientes, el precio, segmento de mercado y cantidad de productos sustitutos.

Los modelos para establecer pronósticos utilizados en este trabajo de grado son **modelos cuantitativos univariados**.

2.2 SERIES DE TIEMPO

Una serie de tiempo es una secuencia ordenada, con intervalos constantemente separados en el tiempo, de valores de una variable.

Las series de tiempo tienen diversas aplicaciones. En el caso del pronóstico, son utilizadas para obtener una comprensión de los factores que producen los datos e intentar identificar un patrón. Una vez identificado el comportamiento y patrón de los datos, se establece y ajusta un modelo para continuar con el proceso de pronóstico.

En los modelos utilizados en este trabajo y con objeto de identificar el patrón de los datos analizados se analiza la serie de tiempo como un conjunto de varios componentes:

1. Tendencia
2. Ciclo
3. Variaciones Estacionales
4. Fluctuaciones Irregulares

La **tendencia** se refiere a la dirección general a la que una serie de tiempo parece dirigirse en un intervalo grande de tiempo.

Los movimientos de la tendencia representan una variedad de factores. Por ejemplo, los movimientos de larga duración en las ventas de una industria en particular podrían estar determinados por un cambio en los gustos de los consumidores o aumento en el ingreso per cápita.

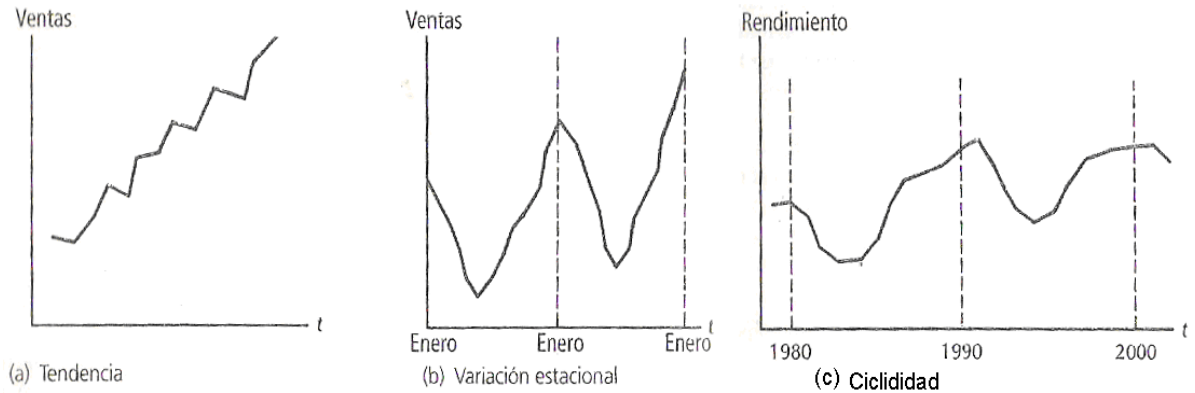
El **ciclo** se refiere a las oscilaciones de larga duración alrededor de la tendencia. Una de las fluctuaciones cíclicas más comunes que se encuentran en las series de tiempo es el “ciclo económico”. Este ciclo se representa por fluctuaciones en las series de tiempo ocasionadas por periodos recurrentes de expansión alternando con recesión.

Las **variaciones estacionales** son patrones periódicos en una serie de tiempo que se completa dentro de un período y se repiten cada período. Factores como el clima y el comportamiento de los consumidores ocasionan variaciones estacionales. Por ejemplo, las ventas de las tiendas por departamentos generalmente alcanzan cantidades altas en diciembre, así como, debido al clima, la ocupación hotelera en zonas costeras suele ser más alta durante el verano.

Las **fluctuaciones irregulares** son “lo que queda” en una serie de tiempo después de que han sido explicados la tendencia, el ciclo y las variaciones estacionales. Representan movimientos esporádicos o de corto plazo en una serie de tiempo que no siguen ningún patrón. Al reconocer este componente, estamos reconociendo que va a haber un error asociado con el modelo de pronóstico.

A continuación se muestran series de tiempo en las que se puede observar la tendencia, variaciones estacionales y componentes cíclicos.

Figura 2-2 Series de tiempo con Tendencia, Estacionalidad y Ciclicidad



2.3 TÉCNICAS DE SUAVIZADO

Las técnicas de suavizado eliminan o alisan las oscilaciones de la serie de tiempo y muestran los componentes de tendencia y ciclo. Inherente a la recolección de datos a lo largo del tiempo hay una forma de variación aleatoria. Existen métodos para reducir el efecto de la variación aleatoria. Una técnica utilizada comúnmente en la práctica es el suavizado. Esta técnica, cuando se usa de manera adecuada, revela más claramente los componentes de tendencia, estacionalidad y ciclo.

Existen dos grupos de métodos de suavizado:

- Métodos de Promedio Móviles
- Métodos de Suavizado Exponencial

Tomar el promedio es la manera más simple de suavizar los datos. Para ilustrar este método se muestra continuación un ejemplo del método de promedio utilizando el promedio simple de los datos históricos de la cantidad de materia prima (en toneladas) entregada por proveedor anualmente.

Se desea conocer la cantidad de materia prima entregada por este proveedor en un año promedio, en toneladas. Para esto se toma una muestra de los últimos 12 años, y se obtienen los siguientes datos:

Tabla 2-1 Toneladas de materia prima por entregadas por proveedor en los últimos años

Año	Toneladas	Año	Toneladas
1	9	7	11
2	8	8	7
3	9	9	13
4	12	10	9
5	9	11	11
6	12	12	10

El promedio o media aritmética de los datos es 10. Se decide utilizar este valor como estimado. ¿Qué tan buena es ésta estimación?

El error cuadrático medio (MSE del inglés *Mean squared error*) es una manera de juzgar qué tan bueno es un modelo. Para esto se computa el error cuadrático medio:

- El *error* es el valor verdadero menos el valor estimado.
- El *error cuadrático* es el error hallado en el paso anterior, al cuadrado.

- SSE es la suma de los errores al cuadrado.
- MSE es la media de los errores al cuadrado.

Estos son los resultados con la media (igual a 10) como estimador:

Tabla 2-2 Error y Error Cuadrático

Año	Toneladas	Error	Error ²
1	9	-1	1
2	8	-2	4
3	9	-1	1
4	12	2	4
5	9	-1	1
6	12	2	4
7	11	1	1
8	7	-3	9
9	13	3	9
10	9	-1	1
11	11	1	1
12	10	0	0

El SSE = 36 y el MSE = $36/12 = 3$.

Para responder la pregunta de qué tan bueno es la media como estimador, se realizan los cálculos con los siguientes valores como estimador: 7, 9, y 12. Es decir, el estimado de que el proveedor entregue en un año promedio 7, 9 o 12 toneladas.

Realizando los mismos cálculos, se encuentra:

Tabla 2-3 Distintos estimadores comparados

Estimador	7	9	10	12
SSE	144	48	36	84
MSE	12	4	<u>3</u>	7

Como estamos utilizando el MSE para medir la calidad el estimador, el estimador con el menor MSE es el mejor.

A continuación se utiliza la media como estimador de las ventas de una empresa de manufactura.

La siguiente tabla muestra los ingresos de un fabricante de computadoras entre 1985 y 1994¹⁴.

¹⁴ <http://www.itl.nist.gov/div898/handbook/pmc/section4/pmc42.htm>

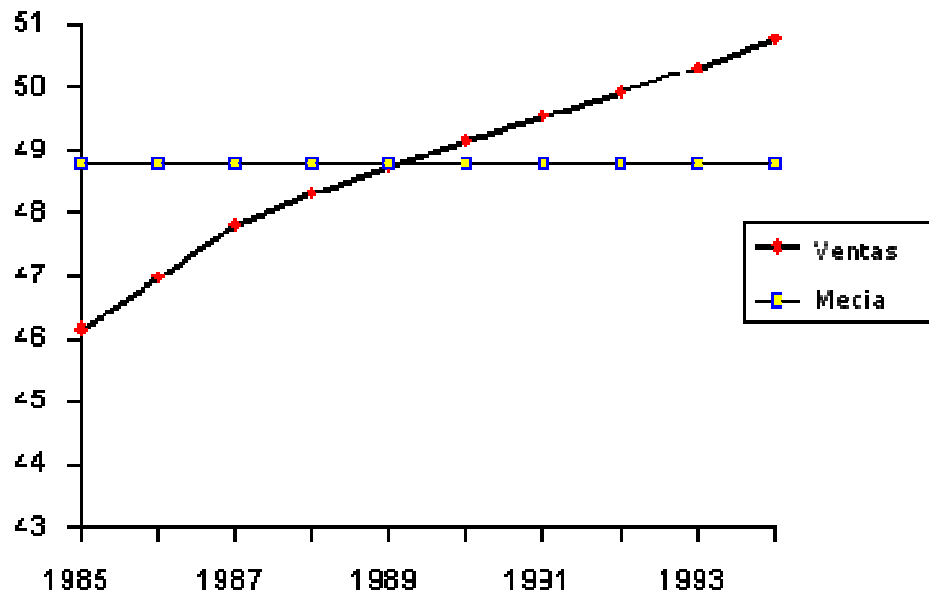
Tabla 2-4 Ventas Fabricante PC 1985 - 1994

Año	\$ (millones)	Media	Error	Error ²
1985	46.163	48.776	-2.613	6.828
1986	46.998	48.776	-1.778	3.161
1987	47.816	48.776	-0.960	0.922
1988	48.311	48.776	-0.465	0.216
1989	48.758	48.776	-0.018	0.000
1990	49.164	48.776	0.388	0.151
1991	49.548	48.776	0.772	0.596
1992	48.915	48.776	1.139	1.297
1993	50.315	48.776	1.539	2.369
1994	50.768	48.776	1.992	3.968

El MSE de los datos en la tabla anterior es igual a 1.9508.

La media no es un buen estimador cuando existe una tendencia. Esto se puede observar claramente en la siguiente gráfica:

Gráfica 2-3 Ventas y Media



Al darle igual peso a todas las observaciones pasadas, la media no es útil para realizar pronósticos cuando los datos tienen un componente de tendencia.

$$\tilde{x} = \frac{1}{n} \sum_{i=1}^n x_i = \left(\frac{1}{n}\right)x_1 + \left(\frac{1}{n}\right)x_2 + \left(\frac{1}{n}\right)x_3 + \dots + \left(\frac{1}{n}\right)x_n$$

La cantidad $\left(\frac{1}{n}\right)$ es el peso. Todos los pesos juntos suman 1.

2.4 MEDIA MÓVIL

Una forma alternativa es calcular el promedio de subconjuntos sucesivos de observaciones pasadas.

El conjunto 9, 8, 9, 12, 9, 12, 11, 7, 13, 9, 11, 10 es la cantidad en toneladas de los 12 años anteriores, mostrados en la Tabla 2-1.

Si se elige N, el tamaño del subconjunto a utilizar, igual a 3. Entonces el promedio de los primeros tres números es: $(9 + 8 + 9) / 3 = 8.667$.

De esta manera también se están suavizando los datos. Este proceso continua avanzando al siguiente periodo y calculando el promedio de los tres números siguientes. La siguiente tabla resume este proceso, conocido como **Media Móvil (MA)**.

La expresión general para la media móvil es:

$$M_t = [X_t + X_{t-1} + \dots + X_{t-N+1}] / N$$

Tabla 2-5 Media Móvil

Año	Toneladas	MA	Error	Error ²
1	9			
2	8			
3	9	8.667	0.333	0.111
4	12	9.667	2.333	5.444
5	9	10.000	-1.000	1.000
6	12	11.000	1.000	1.000
7	11	10.667	0.333	0.111

8	7	10.000	-3.000	9.000
9	13	10.333	2.667	7.111
10	9	9.667	-0.667	0.444
11	11	11.000	0	0
12	10	10.000	0	0

El MSE = 2.018, comparado con MSE = 3 al utilizar la media.

2.5 MEDIA MÓVIL CENTRADA

De ahora en adelante, para reducir el número de cálculos, se toman los valores de la tabla únicamente hasta el año 7.

En el ejemplo anterior se computo la media móvil de los tres primeros periodos y se ubico junto al periodo 3. Se hubiera podido haber ubicado el promedio en el medio del intervalo de tiempo de los 3 periodos, es decir, junto al periodo 2. Esto funciona cuando el periodo (valor de M) es impar. ¿Pero qué ocurre con periodos de tiempo pares? Si $M=4$, técnicamente el promedio móvil estaría en $t = 2.5, 3.5, \dots$

Para evitar este problema, si se utilizan periodos de tiempo pares, se suavizan de nuevo los valores de la media móvil utilizando $M=2$. Ósea que se suavizan los valores suavizados.

A continuación se muestran los resultados al utilizar un periodo par ($M = 4$).

Tabla 2-6 Media Móvil Centrada

Periodo	Valor	MA	MA Centrada
1	9		
1.5			
2	8		
2.5		9.5	
3	9		9.5
3.5		9.5	
4	12		10.0
4.5		10.5	
5	9		10.750
5.5		11.0	
6	12		
6.5			
7	11		

Los resultados finales son:

Tabla 2-7 Tabla Final de MA centrada

Periodo	Valor	MA Centrada
1	9	
2	8	
3	9	9.5
4	12	10.0
5	9	10.75
6	12	
7	11	

2.6 SUAVIZADO EXPONENCIAL

Este método es una manera muy común de producir una serie de tiempo suavizada. Mientras en el método de la Media Móvil las observaciones pasadas tienen igual peso, en el suavizado exponencial las observaciones pasadas se suavizan utilizando pesos exponencialmente decrecientes. Es decir, la observación es menos importante a medida que es más vieja o lo que es igual: las observaciones mas recientes tienen un peso relativamente mayor que las observaciones más viejas.

En el caso de la media móvil, los pesos son iguales a $1/N$. En el suavizado exponencial, sin embargo, hay uno o más parámetros de suavizado a ser

determinados (o estimados) y su elección determina el peso asignado a las observaciones.

En este proyecto se trabajo el suavizado exponencial simple. Método que se describe a continuación:

Este método inicia haciendo S_2 igual a y_1 , donde S_i representa la observación suavizada o EWMA (del inglés *Exponentially Weighted Moving Average*), y y_i representa la observación original.

Los subíndices se refieren a los periodos de tiempo, 1, 2,..., n .

Para el tercer periodo, $S_3 = \alpha y_2 + (1-\alpha)S_2$; y así sucesivamente. No hay S_1 ; la serie suavizada empieza con la versión suavizada de la segunda observación.

Para cualquier periodo de tiempo t , el valor suavizado S_t se encuentra computando:

$$S_t = \alpha y_{t-1} + (1-\alpha)S_{t-1}, \quad 0 < \alpha \leq 1, \quad t \geq 3$$

Esta es la ecuación básica del suavizado exponencial y la constante o parámetro α se conoce como constante de suavizado.

Al expandir la ecuación básica sustituyendo S_{t-1} en la ecuación anterior obtenemos:

$$S_t = \alpha y_{t-1} + (1-\alpha) [\alpha y_{t-2} + (1-\alpha) S_{t-2}] = \alpha y_{t-1} + \alpha(1-\alpha) y_{t-2} + (1-\alpha)^2 S_{t-2}$$

Sustituyendo S_{t-2} , después S_{t-3} , y así sucesivamente, hasta alcanzar S_2 (que no es más que y_1), se puede demostrar que la ecuación puede ser escrita como:

$$S_t = \alpha \sum_{i=1}^{t-2} (1-\alpha)^{i-1} y_{t-i} + (1-\alpha)^{t-2} S_2, \quad t \geq 2$$

Por ejemplo, para el caso del valor suavizado S_5 la ecuación expandida es:

$$S_5 = \alpha \left[(1-\alpha)^0 y_{5-1} + (1-\alpha)^1 y_{5-2} + (1-\alpha)^2 y_{5-3} \right] + (1-\alpha)^3 S_2$$

$\alpha (1-\alpha)^t$ decrecen geométricamente y su suma es 1.

En la siguiente tabla se muestran los pesos para las primeros 4 observaciones de y cuando alfa vale 0.3.

Tabla 2-8 Pesos para $\alpha = .3$

Valor	Peso
y_1	.2100
y_2	.1470
y_3	.1029
y_4	.0720

Por ultimo queda la pregunta ¿Cómo elegir el valor de α ?

La velocidad a la que los datos más antiguos son suavizados es una función del valor de α . Cuando α tiene un valor cercano a 1, los datos son suavizados rápidamente mientras que cuando α tiene un valor cerca de 0, son suavizados lentamente. Esto se ilustra en la siguiente tabla:

Tabla 2-9 Valores de α a medida que t aumenta

α	$(1-\alpha)$	$(1-\alpha)^2$	$(1-\alpha)^3$	$(1-\alpha)^4$
.9	.1	.01	.001	.0001
.5	.5	.25	.125	.0625
.1	.9	.81	.729	.6561

Se debe elegir el valor de α que minimice el MSE.

Este proceso se ilustra mejor tomando los siguientes datos de ejemplo

Tabla 2-10 Error para alfa = .1

Tiempo	y_t	$S(\alpha = .1)$	Error	Error Cuadrático
1	71			
2	70	71	-1.00	1.00
3	69	70.9	-1.90	3.61
4	68	70.71	-2.71	7.34
5	64	70.44	-6.44	41.47
6	65	69.80	-4.80	23.04
7	72	69.32	2.68	7.18
8	78	69.58	8.42	70.90
9	75	70.43	4.57	20.88
10	75	70.88	4.12	16.97
11	75	71.29	3.71	13.76
12	70	71.67	-1.67	2.79

En este caso, SSE = 208.94. Y el MSE igual a: SSE /11 = 19.0

Si se utiliza un valor de $\alpha = 0.5$ el MSE resulta ser 16.29, lo cual es mejor que 19.0. De manera que se preferiría utilizar un valor de 0.5. Para determinar el mejor valor de α , se podría mediante un proceso iterativo intentar con todos los valores de α entre $\alpha - \Delta$ y $\alpha + \Delta$ para encontrar el valor que minimice el MSE. Sin embargo existen métodos numéricos no lineales más eficientes que se utilizan en

la herramienta software para realizar este cálculo. En particular se utiliza el método de Nelder-Mead¹⁵ implementado en la librería Scipy^{16, 17}.

En la Figura 2-11 se muestra la serie original y los valores suavizados para dos valores de alfa distintos.

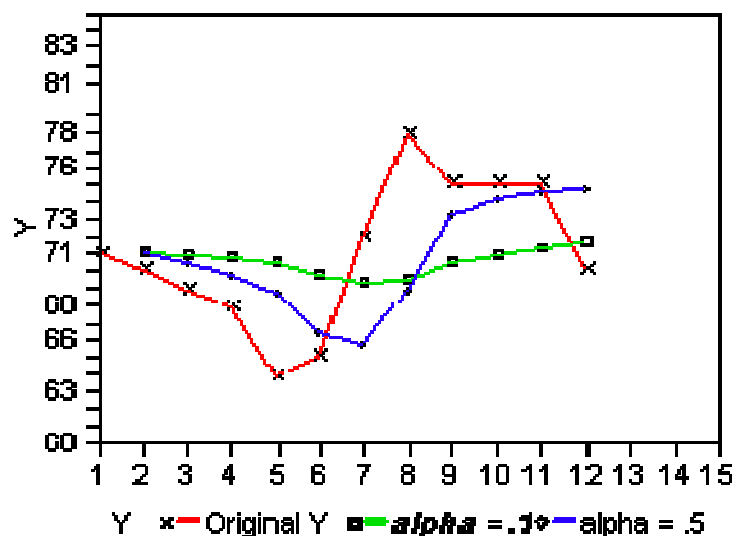


Figura 2-11 Serie Original y Suavizado Exponencial

En el Anexo B se puede encontrar el código fuente que realiza el suavizado exponencial simple. Este código puede ser utilizado por cualquier aplicación que importe el módulo estadística.py (incluido y desarrollado junto con la herramienta software)

¹⁵ J. A. Nelder and R. Mead, "A simplex method for function minimization", Computer Journal, 1965, vol 7, pp 308–313

¹⁶ Ver Scipy Sitio Oficial[<http://www.scipy.org>]

2.7 DESCOMPOSICIÓN MULTIPLICATIVA

La aplicación de los **modelos de descomposición** para pronosticar series temporales se apoya en la idea básica de descomponer la serie temporal en factores de: tendencia, estacionalidad, ciclicidad e irregularidad. El uso del **modelo de descomposición multiplicativa** es útil cuando la serie manifiesta tendencia y variaciones estacionales crecientes o decrecientes.

A continuación se presenta una definición formal del modelo:

Considere una serie temporal que manifiesta variación estacional creciente o decreciente. Cuando los parámetros que describen la serie no cambian en el tiempo, a veces, la serie temporal se puede modelar en forma adecuada usando el **modelo de descomposición multiplicativa**

$$y_t = TR_t \times SN_t \times CL_t \times IR_t$$

Donde

y_t = valor observado de la serie temporal en el periodo t

TR_t = el componente (o factor) de la tendencia en el periodo t

SN_t = el componente (o factor) estacional en el periodo t

CL_t = el componente (o factor) cíclico en el periodo t

IR_t = el componente (o factor) irregular en el periodo t

Para mostrar el uso del modelo se presenta el ejemplo resuelto 12.2 propuesto en Sarmiento [2], reproducido textualmente con autorización del autor.

Ejemplo 12.2

Ventas trimestrales del R12, dadas en la siguiente página en las columnas 1 y 3. Se desea establecer un modelo predictivo (que permite hacer una predicción sobre el comportamiento futuro del sistema).

P1. Escogemos el modelo multiplicativo: $Y = T \cdot C \cdot E \cdot I$, Tendencia (refleja el efecto de los factores que afectan el comportamiento a largo plazo) * Ciclicidad (refleja el efecto de los factores que afectan el comportamiento en forma cíclica, en períodos diferentes al anual) * Estacionalidad (refleja el efecto de los factores que afectan el comportamiento en ciclos de un año, que son los más comunes) * Incontrolado (refleja el efecto de otros factores que afectan el comportamiento, pero que o no los conocemos, o no se habían presentado antes, o no obedecen a algún patrón, por lo que ese efecto es “aleatorio”), y se asigna en orden, del 1 al 32 a cada trimestre (col. 2)

P2. Establecemos el promedio móvil para eliminar las variaciones estacionales e incontrolado, tomando el promedio de tres datos, dos antes y uno después, en la Col. 4. Se centra ese promedio como el promedio de dos datos de la columna 4, el de este trimestre y el siguiente, en la Col. 5

P3. Para la tendencia T, la asumimos lineal: $T = a + Bx$

$$n = 32$$

$$\sum X = 528$$

$$\sum Y = 12.480$$

$$\sum XY = 223.229$$

$$\sum X^2 = 11.440$$

$$b = \frac{\frac{\sum XY}{n} - \overline{XY}}{\frac{\sum X^2}{n} - \overline{X}^2} = \frac{\frac{223229}{32} - \frac{528}{32} * \frac{12480}{32}}{\frac{11440}{32} - \left(\frac{528}{32}\right)^2} = 6.34 \text{ carros/trimestre}$$

$$a = \overline{Y} - b\overline{X} = \frac{12480}{32} - 6.34 * \frac{528}{32} = 285.39 \text{ carros}$$

- P4. Con los anteriores estimadores establecemos la tendencia T, resultando la Col. 7
- P5. Se calcula el Estacionalidad por el factor Incontrolado, dividiendo Col. 3 por Col. 5, en Col. 6
- P6. Calculamos la Estacionalidad promediando cada trimestre de la Col. 6 para eliminar el factor Incontrolado, y se ajusta a la unidad. Así pasamos al cuadro al final de la tabla en el 1^{er} dato el promedio de todos los trimestres uno (sumamos 76.28 correspondiente a 1/74, 81.07 del 1/75,... y luego dividimos por 7, el total de sumandos, dando el 79.73 del 1er trimestre), el promedio del trimestre dos, el tres y el cuatro. Como el promedio no dio 100% como se esperaba, verificamos que ese error no sea grande (se debió a que no le dimos igual peso a todos los datos, ya que los primeros y los últimos tienen menos peso que los demás), si lo es quiere decir que no tenemos suficientes datos para ese método y habrá que buscar uno menos preciso; si el error es pequeño lo ajustamos, dividiendo cada dato por el resultante en por unidad (en nuestro caso resultó 99.475%, entonces dividimos cada promedio de estacionalidad trimestral -el promedio del trimestre- por 0.99475), y los nuevos promedios tendrán un promedio ajustado.

Se copian esos valores en el trimestre respectivo, Col 8

P7. Dividiendo Col 3 por Col 7 y Col 8, obtenemos la Ciclicidad por el Incontrolado, Col 9

P8. Promediamos la Col 9 por tres trimestres, uno antes, el de ese trimestre y uno después, para suavizar eliminando el factor Incontrolado, aislando en esta forma la Ciclicidad, Col 10

La tabla completa se muestra en la siguiente página.

Tabla 2-12 Datos del Ejemplo. 12.1

(1) Trim	(2) Tri	(3) Vnt Aut	(4) Prom Movil	(5) Centr Autos	(6) Y/T*C p.u.	(7) Autos	(8) E p.u.	(9) p.u.	(10) Prom. p.u.	(11) p.u.	(12) Lm.I Auto	(13) Lm.S Auto	(14) Err- Aut	(15) Err Aut	(16) Err %
	X	Y		=T*C	=E*I	=T		=C*I	=C	=I					
III/73	1	398				291,7	1,076	1,267			352,	405,6	0	0	0,0
IV/73	2	352				298,0	0,911	1,296	1,241	1,044	313,	360,7	0	0	0,0
I/74	3	283	371,7	371,0	0,762	304,4	0,801	1,159	1,220	0,950	277,	318,7	0	0	0,0
II/74	4	454	370,2	369,3	1,229	310,7	1,211	1,206	1,171	1,029	410,	471,9	0	0	0,0
III/74	5	392	368,5	367,3	1,067	317,0	1,076	1,148	1,175	0,977	373,	429,1	0	0	0,0
IV/74	6	345	366,2	358,5	0,962	323,4	0,911	1,170	1,118	1,046	306,	352,8	0	0	0,0
I/75	7	274	350,7	338,0	0,810	329,7	0,801	1,036	1,056	0,981	260,	298,9	0	0	0,0
II/75	8	392	325,2	308,3	1,271	336,1	1,211	0,962	0,928	1,036	352,	404,6	0	0	0,0
III/75	9	290	291,5	284,5	1,019	342,4	1,076	0,786	0,803	0,979	275,	316,9	0	0	0,0
IV/75	10	210	277,5	276,2	0,760	348,7	0,911	0,660	0,737	0,895	218,	250,9	8,3	0	3,9
I/76	11	218	275,0	286,5	0,760	355,1	0,801	0,765	0,766	0,999	203,	233,4	0	0	0,0
II/76	12	382	298,0	314,2	1,215	361,4	1,211	0,872	0,867	1,005	353,	406,6	0	0	0,0
III/76	13	382	330,5	340,5	1,121	367,8	1,076	0,965	0,945	1,021	348,	400,3	0	0	0,0
IV/76	14	340	350,5	359,2	0,946	374,1	0,911	0,997	0,979	1,017	311,	357,4	0	0	0,0
I/77	15	298	368,0	373,1	0,798	380,4	0,801	0,977	0,979	0,997	278,	319,7	0	0	0,0
II/77	16	452	378,2	382,2	1,182	386,8	1,211	0,964	0,980	0,983	427,	491,6	0	0	0,0
III/77	17	423	386,2	391,0	1,081	393,1	1,076	0,999	0,995	1,004	392,	450,7	0	0	0,0
IV/77	18	372	395,7	397,7	0,935	399,5	0,911	1,022	1,018	1,003	345,	396,6	0	0	0,0
I/78	19	336	399,7	395,2	0,850	405,8	0,801	1,033	0,997	1,035	302,	347,2	0	0	0,0
II/78	20	468	390,7	382,8	1,222	412,1	1,211	0,937	0,943	0,993	438,	503,9	0	0	0,0
III/78	21	387	375,0	366,0	1,057	418,5	1,076	0,859	0,864	0,993	362,	416,9	0	0	0,0
IV/78	22	309	357,0	348,3	0,887	424,8	0,911	0,798	0,807	0,989	290,	334,3	0	0	0,0
I/79	23	264	339,7	342,3	0,771	431,2	0,801	0,763	0,771	0,989	248,	285,4	0	0	0,0
II/79	24	399	345,0	355,8	1,121	437,5	1,211	0,752	0,790	0,952	389,	448,2	0	0	0,0
III/79	25	408	366,7	382,3	1,067	443,8	1,076	0,854	0,857	0,996	381,	438,3	0	0	0,0
IV/79	26	396	398,0	423,6	0,934	450,2	0,911	0,965	0,960	1,004	366,	421,7	0	0	0,0
I/80	27	389	449,2	470,6	0,826	456,5	0,801	1,063	1,035	1,026	352,	405,4	0	0	0,0
II/80	28	604	492,0	506,6	1,192	462,9	1,211	1,077	1,095	0,983	571,	657,3	0	0	0,0
III/80	29	579	521,2	536,3	1,079	469,2	1,076	1,146	1,135	1,009	534,	613,8	0	0	0,0
IV/80	30	513	551,5	558,6	0,918	475,5	0,911	1,183	1,216	0,972	490,	564,3	0	0	0,0
I/81	31	510	565,7			481,9	0,801	1,320	1,207	1,093	434,	499,0	0	11,	2,2
II/81	32	661				488,2	1,211	1,117	1,241		683,	785,5	22,	0	3,4
III/81	33					494,6	1,076		1,220		604,	695,4			
Medi	16,	390	381,2	378,1	0,994	390	1,00	1,004	0,989	1,000				Σ=	9,5
Des.	9,3	104	73,13	69,12	0,164	59,48	0,158	0,170	0,151	0,035					

En la tabla los datos en bastardilla son estimados, no se consideraron para el promedio ni la desviación estándar. El error % total no es promedio, es la suma de los errores.

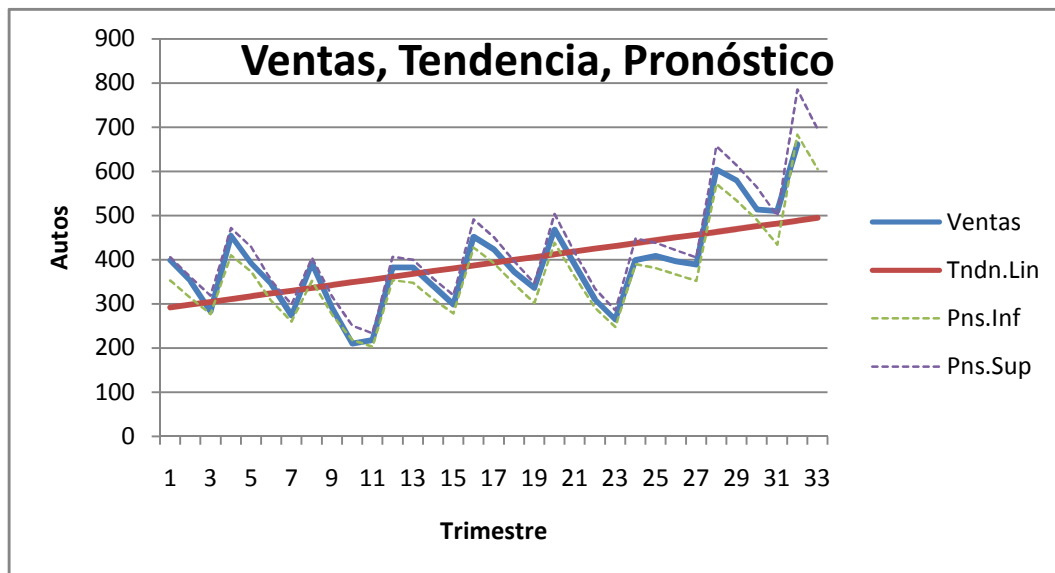
P9. Conocidos los factores T, E y C (en Col 7, 8 y 10), dividimos el total de Col 3 por las anteriores, resultando el factor Incontrolado, que resulta ser el único aleatorio del modelo.

P10. Se grafican las columnas 3 y 7 contra la Col 2

P11. Se grafican las columnas 10 y 11 contra la Col 2

Tabla 2_12. Cálculo de la Estacionalidad, %:

	I	II	III	IV	
1974	76,28%	122,91%	106,70%	96,23%	
1975	81,07%	127,12%	101,93%	76,02%	
1976	76,09%	121,56%	112,19%	94,64%	
1977	79,87%	118,25%	108,18%	93,53%	
1978	85,01%	122,23%	105,74%	88,70%	
1979	77,11%	112,12%	106,70%	93,48%	
1980	82,66%	119,22%	107,95%	91,83%	
Prom	79,725%	120,487%	107,056%	90,633%	99,475%
Ajust	80,146%	121,122%	107,621%	91,111%	100,000%



Se anexan las gráficas:

- Arriba, las ventas (Col. #3) y la tendencia lineal (Col. #7) que se supuso, el pronóstico inferior (Col #12), pronóstico superior (Col #13), y en la abscisa el trimestre (Col #2). Los dos límites del pronóstico se calcularon con un 95% de confianza. Nótese cómo las ventas se salen de los límites calculados sólo en los trimestres #10 y #36 por debajo, y el trimestre #31 por encima.
- Abajo, la Ciclicidad (Col. #10), donde se aprecia que el primer punto ($X=2$) tiene un valor similar al último ($X=31$), por lo que se puede pensar que se sigue repitiendo. En esa misma gráfica se incluye el factor Incontrolado (Col. #11), que se supone es el único aleatorio, y en el que se notan muy cercanos al 100%, excepto en el trimestre #10, el #31 y el #32.
- Abajo, dos gráficas de la Col. #6 (Estacionalidad*Incontrolado) para observar qué tanto se repite el ciclo Estacional propuesto. En la primera

se muestra por año donde se aprecia la similitud entre los diferentes años; la mayor diferencia la encontramos en el año 1978, pero es pequeña. En la segunda se muestra por trimestre, allí se aprecia que para cada trimestre el valor es muy uniforme para todos los años; siendo el punto más desviado el IV Trimestre de 1975.

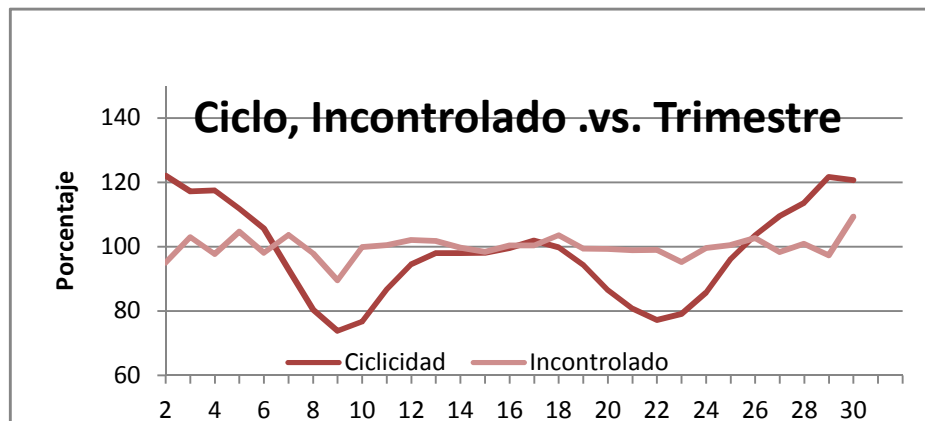


Ilustración 0-4 Ciclo, Incontrolado Vs. Trimestre

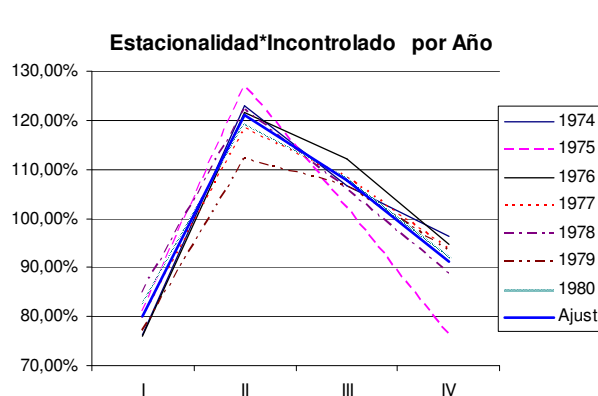


Ilustración 0-5 Estacionalidad x Incontrolado Anual

Como verificación, predecimos cada uno de los trimestres incluyendo el siguiente. La mayoría de los errores se concentran al final, puede deberse en el penúltimo a que se iba a acabar, y en el último a que ya no se produjeron los suficientes.

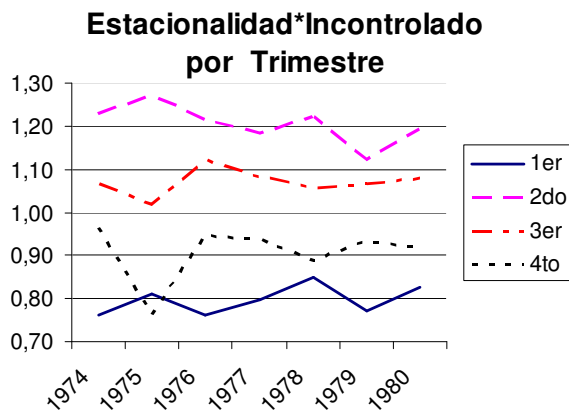


Figura 2-6 Estacionalidad x Incontrolado Semestral

Este ejercicio junto con sus cálculos e ilustraciones pueden ser resuelto utilizando la herramienta software desarrollada. Ver capítulo 4: 4. SOLUCIÓN Y RESULTADOS

2.8 PROGRAMACIÓN ORIENTADA A OBJETOS¹⁸

El desarrollo de programas ha pasado de ser una curiosidad a ser una de las industrias más importantes en medio siglo. Esto solo fue posible cuando se evidenció la necesidad de utilizar el código desarrollado anteriormente en vez de caer en el síndrome de re-inventar la rueda y se establecieron mecanismos

¹⁸ Akimuskín Valencia, Camilo. Interfaz de usuario gráfica para programa de elementos finitos. Proyecto de Grado Ingeniero Mecánico. Universidad Industrial de Santander, 2006

eficientes para permitir a los programadores manejar el código importado. El último y mas importante de estos mecanismos es el paradigma de la programación orientada a objetos (*Object Oriented Programming, OOP*) el cual no ha sido destronado durante más de dos décadas.

La OOP define las reglas que se deben seguir al utilizar código en un nuevo programa y al escribir código para que sea re-utilizado por otros, así que será necesaria al trabajar con el código escrito.

La programación orientada a objetos le permite a una persona desarrollar programas de cientos de miles de líneas de código en lenguaje de alto nivel gracias a la re-utilización de código anterior. La re-utilización de código tiene la doble ventaja de la modularidad y de evitar los errores ocultos basándose en la confianza en el código reutilizado, el cual en lo posible, debe no modificarse.

La sola definición de la programación orientada a objetos es muy abstracta y sujeta a diferentes interpretaciones, basta con tener en cuenta que el código escrito con éste paradigma esta organizado en unidades fundamentales llamadas objetos (que pueden ser clases o estructuras).

Los objetos contienen variables y funciones llamadas miembros del objeto, que solo se pueden acceder si se declaran como *public*, los miembros que se declaran como *protected* o *private* solo se pueden acceder por otros miembros de la clase o sus clases descendientes, lo cual se conoce como encapsulado de la información.

La programación orientada a objetos se fundamenta en dos pilares, la herencia y el polimorfismo.

Con el primero se pueden definir clases hijas o descendientes de otras clases padre o base creando una vasta genealogía de objetos con propiedades comunes, llamada jerarquía.

Con el segundo se pueden escoger funciones en tiempo de ejecución utilizando funciones declaradas virtuales en la clase base y volviéndolas a definir en las clases descendientes.

2.9 INTERFACES DE USUARIO¹⁹

La comunicación entre una persona y una computadora, como cualquier comunicación requiere esencialmente de cuatro elementos indispensables:

- Emisor: aquel que emite la información.
- Receptor: aquel que recibe la información.
- Canal: medio a través del cual la información es transmitida.
- Código: lenguaje común a ambos interlocutores.

¹⁹ Akimuskín Valencia, Camilo. Interfaz de usuario gráfica para programa de elementos finitos. Proyecto de Grado Ingeniero Mecánico. Universidad Industrial de Santander, 2006

El código o lenguaje utilizado va a depender de la interfaz de usuario manejada, y puede ser más o menos rico en significado, sencillo o adaptable. Cada programa ofrece su propia interfaz incluyendo a los sistemas operativos. Los programas que funcionan bajo un sistema operativo normalmente utilizan la interfaz ofrecida por el mismo.

Las interfaces de usuario se encargan de manejar los canales que tengan relación con el usuario (teclado, ratón, pantalla, impresora, etc.) usando los respectivos códigos comunes al programa y al usuario. Existen otras interfaces que manejan la comunicación del programa con otros dispositivos, afortunadamente los sistemas operativos también pueden manejar estas interfaces por lo cual no se hace necesario que el programa se relacione directamente con los dispositivos sino con el sistema operativo. En definitiva, el programa debe comunicarse solamente con el usuario y con el sistema operativo, esta comunicación se logra a través de eventos que son manejados mediante mensajes.

2.10 INTERFACES GRÁFICAS DE USUARIO

El termino interfaz de usuario gráfica, traducción del inglés *Graphic User Interface*, o GUI fue acuñado como contraposición de las interfaces de usuario por línea de comandos, o CUI. En las CUI los canales de entrada y salida de información son flujos de cadenas de caracteres (*standard input/output stream*) mientras que en las GUI los canales de entrada son ratones, joysticks, etc. además del teclado y la salida en pantalla no solo es texto, también incluye elementos gráficos como menús, ventanas, botones, etc. Actualmente éstas dos son las interfaces de usuario mas utilizadas.

Si bien un programa puede ofrecer la interfaz que quiera mientras sea el programa seleccionado con el foco, la interfaz de usuario de un programa va a estar limitada por la interfaz de usuario del sistema operativo, ya que éste también es un programa quien controla a los dispositivos periféricos. Antes, existían programas que tenían una interfaz de usuario (gráfica) independiente del SO y mas sofisticada, algunos programas (como los juegos) aun conservan esta practica pero como se verá, ahora que los SO han evolucionado, se prefiere una mayor integración con el escritorio.

2.11 GUÍAS Y RECOMENDACIONES DE DISEÑO

Según algunas organizaciones especializadas²⁰, existen ciertas recomendaciones que pueden seguir aquellos que desarrollan nuevos programas, relacionadas con el funcionamiento, interacción con el usuario, manejo de datos, etc. Estos son algunos de los parámetros con los cuales se mide el desempeño de un programa.

2.11.1 Accesibilidad

La accesibilidad significa permitir a una persona con algún tipo de discapacidad participar de las actividades de la vida. Al utilizar un software, los colores son inútiles para distinguir información para los usuarios daltónicos, igualmente lo son los sonidos para los sordos y los usuarios con movilidad limitada no pueden usar una aplicación sin las adecuadas equivalencias en el teclado. Una aplicación verdaderamente accesible debe permitir además, lectores de voz, lectores de pantalla y dispositivos de entrada/salida alternos.

²⁰ GNOME – Human Interface Guidelines

2.11.2 Internacionalización y localización

La internacionalización significa diseñar aplicaciones que funcionen bajo diferentes entornos de lenguaje. La localización se refiere al acto de traducir los mensajes y símbolos a otro idioma. En particular se facilita la tarea de localización si se utilizan recursos como las tablas de cadenas de caracteres.

2.11.3 Respuesta

Es la rapidez con que responde el programa ante una solicitud (mensaje) del usuario.

2.11.4 Entrada del usuario

Se refiere a la comodidad con la cual debe comunicarse el usuario al introducir información al programa, esto se debe tener en cuenta especialmente si el programa va a ser utilizado muchas veces.

2.11.5 Otras recomendaciones²¹

- Evitar los cuadros de diálogo modales, ya que con ellos el usuario solo tiene dos opciones, los botones OK y Cancel. Nunca utilizar cuadros de diálogo modales de sistema (System Modal) ya que bloquean todo el sistema operativo.
- No utilizar alertas sonoras en general.
- Utilizar los aceleradores estandarizados para las tareas comunes en vez de definir o traducir otras combinaciones de teclas.

²¹ GNOME Human Interface Guidelines 2.2.1. Internet. Disponible en <http://library.gnome.org/devel/hig-book/stable/> Accedido noviembre 14 de 2010.

- Cuando se agrupan diferentes opciones en un cuadro de diálogo, debe existir un botón aplicar para cada grupo.

Una aplicación diseñada para funcionar en un sistema operativo gráfico puede contener elementos accesorios, se recomienda utilizar principalmente dos elementos: Una entrada en el menú aplicaciones (o botón de inicio en Windows) y una entrada en el registro de documentos del sistema para que la aplicación se ejecute automáticamente cada vez que el usuario escoja un documento con una extensión del tipo apropiado. Se recomienda no poner ningún icono en el escritorio.

En la entrada en el menú se debe poner el nombre de la aplicación y una pequeña descripción funcional de su comportamiento a menos que el nombre de la aplicación sea lo suficientemente descriptivo. Se recomienda poner un *tooltip* (comentario) para cada entrada en el menú, el cual puede ser un poco más extenso y descriptivo.

Otros elementos pueden ser necesarios en sistemas operativos particulares, por ejemplo, las

GConfKeys usadas en GNOME.

2.12 SISTEMAS OPERATIVOS²²

Los sistemas operativos (en adelante llamados SO) son los programas que se encargan de administrar recursos y permisos en una computadora para permitirle

²² Akimuskín Valencia, Camilo. Interfaz de usuario gráfica para programa de elementos finitos. Proyecto de Grado Ingeniero Mecánico. Universidad Industrial de Santander, 2006

a los usuarios utilizar los demás programas y datos en general sin tener que tratar directamente con los procesos que se llevan a cabo en el equipo utilizado. Además un SO suele incluir no solo utilidades agregadas sino también una interface de programación de aplicaciones (API), diseñada para servir de ayuda a los programas que se ejecutan en el SO.

Existe una gran variedad de sistemas operativos como el Windows de Microsoft, El MacOS de Apple, El OS2 de Sun Microsystems, Inc., el UNIX desarrollado en Bell Labs o el Linux ya sea alguna de las muchas distribuciones gratuitas o de una de las cada vez menos distribuciones pagadas. Debido a que este programa esta diseñado para correr en múltiples plataformas o sistemas operativos, vale la pena recordar la historia de los mismos.

El SO UNIX puede considerarse como el primero usado masivamente, empezó a desarrollarse en los laboratorios Bell Labs en 1969. Antes de UNIX, cada computadora (*mainframe*) tenía su propio SO que los usuarios debían aprender y peor aún, los programas solamente funcionaban en una computadora. UNIX resolvió este problema ya que era simple y sofisticado, capaz de reciclar código y estaba escrito en lenguaje C de alto nivel. Gracias a esto se pudieron crear sistemas operativos UNIX para prácticamente cualquier computadora.

El MS-DOS fue presentado en 1984 por la empresa Microsoft y consistía en una interface por línea de comandos sobre la cual se desarrollarían los siguientes sistemas operativos, Windows 3.1, Windows 95 y Windows 98, paralelamente, Microsoft ofrecía Windows NT para servidores de Internet. Luego aparecerían los sistemas operativos Windows XP y Windows Vista y Windows 7.

El SO LINUX fue creado por Linus Benedict Torvalds a comienzos de los años 90s como una versión académica gratuita para computadoras personales de UNIX. Ya existían versiones comerciales de UNIX para cualquier hardware incluyendo computadoras personales pero eran muy costosas y muy lentas. La popularidad de LINUX hizo que mucha gente contribuyera al proyecto y a los pocos años convirtió a LINUX en una réplica de UNIX con toda su funcionalidad y capaz de utilizarse en un creciente rango de maquinas.

3. DISEÑO METODOLÓGICO

3.1 *METODOLOGÍA DE DESARROLLO*

Para el desarrollo de este software se eligió seguir el paradigma de desarrollo de software basado en procesos ágiles. Los criterios principales al elegir este paradigma son: el ambiente en el que se usa la herramienta software, y evitar la excesiva burocracia y documentación de las metodologías de desarrollo tradicionales, para enfocarse en las necesidades de los usuarios.

El ambiente en el que se utilizará el software es complejo, ya que se desea poner a disposición del estudiante una herramienta con distintos modelos de pronóstico, modelos que a su vez tienen distintos parámetros variables.

La metodología de desarrollo de software utilizada es una metodología ágil, con esto se busca maximizar la comunicación “cara a cara” con los usuarios y demás personas involucradas en el desarrollo de la aplicación.

En las metodologías ágiles, el software desarrollado en una unidad de tiempo es llamado una iteración, la cual debe durar de una a cuatro semanas. Cada iteración del ciclo de vida incluye: planificación, análisis de requerimientos, diseño, codificación, revisión y documentación. Una iteración no debe agregar demasiada funcionalidad para justificar el lanzamiento del producto al mercado, pero la meta

es tener un demo (sin errores) al final de cada iteración. Al final de cada iteración el equipo vuelve a evaluar las prioridades del proyecto.²³

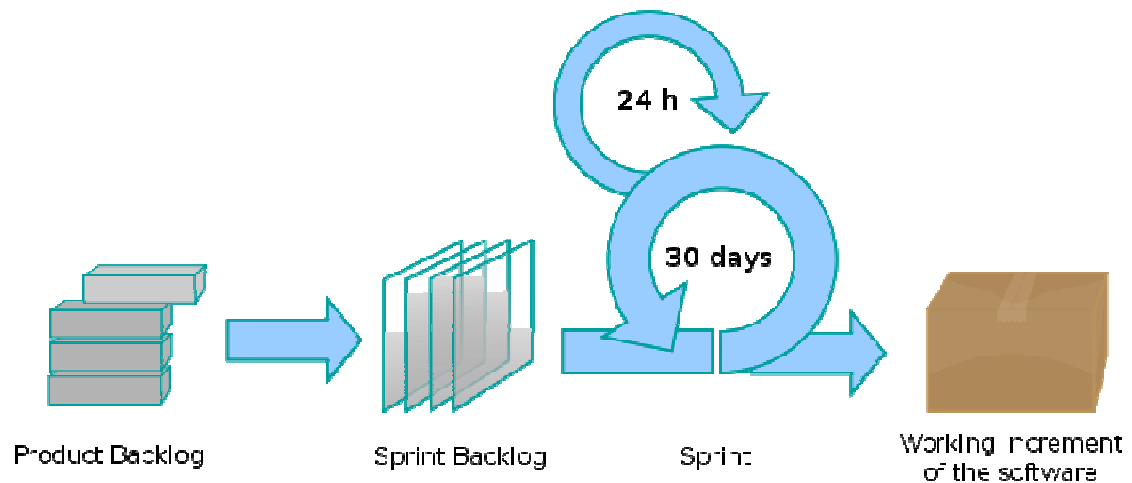


Figura 3-1: Proceso Scrum

Para el proceso de desarrollo de la aplicación y establecer el plan de trabajo, se utilizó el marco de la metodología *Scrum*, adaptado a un equipo de un solo desarrollador.

Obviamente, al ser un equipo de desarrollo de solo dos miembros, no se pudieron llevar a cabo varias de las prácticas de la metodología. En la siguiente sección se muestran los roles y artefactos que se conservan.

²³ Desarrollo ágil de software. (2010, 3) de agosto. Wikipedia, La enciclopedia libre. Fecha de consulta: 13:44, agosto 12, 2010 desde http://es.wikipedia.org/w/index.php?title=Desarrollo_%C3%Agil_de_software&oldid=39252105

En la Figura 3'1 se muestra el esqueleto del proceso *Scrum*.

Para un resumen más detallado del proceso, sus roles y componentes ver Anexo A

3.2 FASES DEL DESARROLLO

De acuerdo a la metodología de desarrollo escogida, se presentan a continuación las principales fases del proyecto y el trabajo realizado en cada una:

3.2.1 Estudio Previo y Recopilación de Información

Durante esta fase se estudió y recopiló el material bibliográfico pertinente a la temática desarrollada, para este caso, Pronóstico y Series de Tiempo junto con Programación en Python y las librerías *pyqt* y *matplotlib*; se determinaron los modelos a tratar y la forma de presentación de los mismos.

3.2.2 Planeación Proyecto y Definición de Metodología

El objetivo de esta etapa es definir los detalles de metodología de desarrollo, darla a conocer a todas las personas involucradas en el proyecto y establecer un plan de trabajo detallado y de común acuerdo para esta fase.

El plan de trabajo esta basado en el marco de proceso *Scrum*, mencionado en el numeral anterior, con varias modificaciones. La duración elegida para un Sprint fue de dos semanas.

De los roles originales, se mantuvieron:

- Equipo
- Usuarios
- Dueño del producto

De las reuniones,

- Reunión de Planificación del Sprint
- Reunión de Revisión del Sprint
- *Daily Scrum*, se reemplazó por una reunión semanal para revisar el estado del proyecto, y si hay algún impedimento que deba ser solucionado.

3.2.3 Desarrollo

Durante la fase de desarrollo se realizaron incrementos al software, siguiendo la metodología mencionada en el numeral anterior. Después de cada iteración, se contó con un producto potencialmente entregable que añadía cierta funcionalidad a la herramienta software.

Los dos primeros Sprints estuvieron enfocados en el desarrollo de la funcionalidad correspondiente al modelo multiplicativo, y los restantes al suavizado exponencial.

Gracias a la metodología empleada, se fueron realizando pruebas a lo largo del proyecto que permitieron ir depurando errores e ir haciendo correcciones a cada uno de los componentes contemplados para el desarrollo del software educativo.

Fue de vital importancia la realimentación obtenida de la funcionalidad implementada en cada iteración para mejorar continuamente el producto. Este proceso se realizó con un grupo de estudiantes de prueba, tomados de la población, para ir verificando que se cumpliera con aspectos de interfaz y funcionalidad requeridas. Ver Capítulo 4. Pruebas para ver una forma en que se adaptaron las pruebas de usabilidad iniciales.

3.2.4 Documentación

La documentación del sistema se realizó durante todo el ciclo de vida del desarrollo y a medida que se entregaba nueva funcionalidad en cada Sprint.

Las tareas desarrolladas en la documentación estuvieron enfocadas en facilitar información detallada acerca del manejo de la herramienta en general y especificando lo relacionado con cada modelo de pronóstico.

El código fuente de la aplicación, también está debidamente documentado.

3.2.5 Entrega y Soporte

Se hará entrega del producto terminado para su uso en un ambiente de estudio “real”. Durante la duración de esta fase se dará soporte al docente en cuanto a la instalación y uso de la aplicación durante el final del segundo periodo académico de 2010.

4. SOLUCIÓN Y RESULTADOS

El resultado de este proyecto es un software de acuerdo a lo planteado en la sección 1.1 OBJETIVOS.

A continuación se presenta este software, su marco de desarrollo, el código fuente así como las clases y módulos que lo conforman. Mediante la resolución de un ejercicio de pronóstico, se muestra su funcionamiento.

4.1 HERRAMIENTAS DE DESARROLLO

La aplicación está desarrollada utilizando principalmente cuatro herramientas, un lenguaje de programación, y tres librerías de programación. A partir de estos cuatro bloques, se realizan los cálculos, se interactúa con el usuario, y se muestran los datos de manera grafica e interactiva.

4.1.1 Lenguaje de Programación: Python

Python es un lenguaje de programación de alto nivel cuya filosofía hace hincapié en una sintaxis muy limpia y que favorezca un código legible.

Se trata de un lenguaje de programación multiparadigma ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación

funcional. Es un lenguaje interpretado, usa tipado dinámico, es fuertemente tipado y es multiplataforma.²⁴

Es administrado por la *Python Software Foundation* y posee una licencia de código abierto

4.1.2 Interfaz Gráfica: librería Qt

Qt es una biblioteca multiplataforma para desarrollar interfaces gráficas de usuario y también para el desarrollo de programas sin interfaz gráfica como herramientas de la consola y servidores. Qt es utilizada principalmente en Autodesk, KDE, Google Earth, la Agencia Espacial Europea, Skype, Qt Extended, Adobe Photoshop Album, VirtualBox, VLC media player, Samsung, Philips, Panasonic y Opie. Es producido por la división de software Qt de Nokia, que entró en vigor después de la adquisición por parte de Nokia de la empresa noruega Trolltech, el productor original de Qt, el 17 de junio de 2008.

Qt es utilizada en KDE, un entorno de escritorio para sistemas como GNU/Linux o FreeBSD, entre otros. Qt utiliza el lenguaje de programación C++ de forma nativa, adicionalmente puede ser utilizado en varios otros lenguajes de programación a través de bindings.²⁵ Debido a que la aplicación esta desarrollada utilizando el lenguaje Python, se utilizó la librería PyQt.

PyQt es un *binding* de la biblioteca gráfica Qt para el lenguaje de programación Python. La biblioteca está desarrollada por la firma británica Riverbank Computing

²⁴ Python. Wikipedia, La enciclopedia libre. Fecha de consulta: 11:37, octubre 20, 2010 de <http://es.wikipedia.org/wiki/Python>.

²⁵ Biblioteca Qt. Wikipedia, La enciclopedia libre. Fecha de consulta: 13:44, octubre 20, 2010 de [http://es.wikipedia.org/wiki/Qt_\(biblioteca\)](http://es.wikipedia.org/wiki/Qt_(biblioteca)).

y está disponible para Windows, GNU/Linux y Mac OS X. Qt, así como PyQt, es software libre y de código abierto.

Se utilizó la herramienta *Qt Designer* para el diseño y generación del código de la interfaz gráfica de la herramienta software implementada en este proyecto.

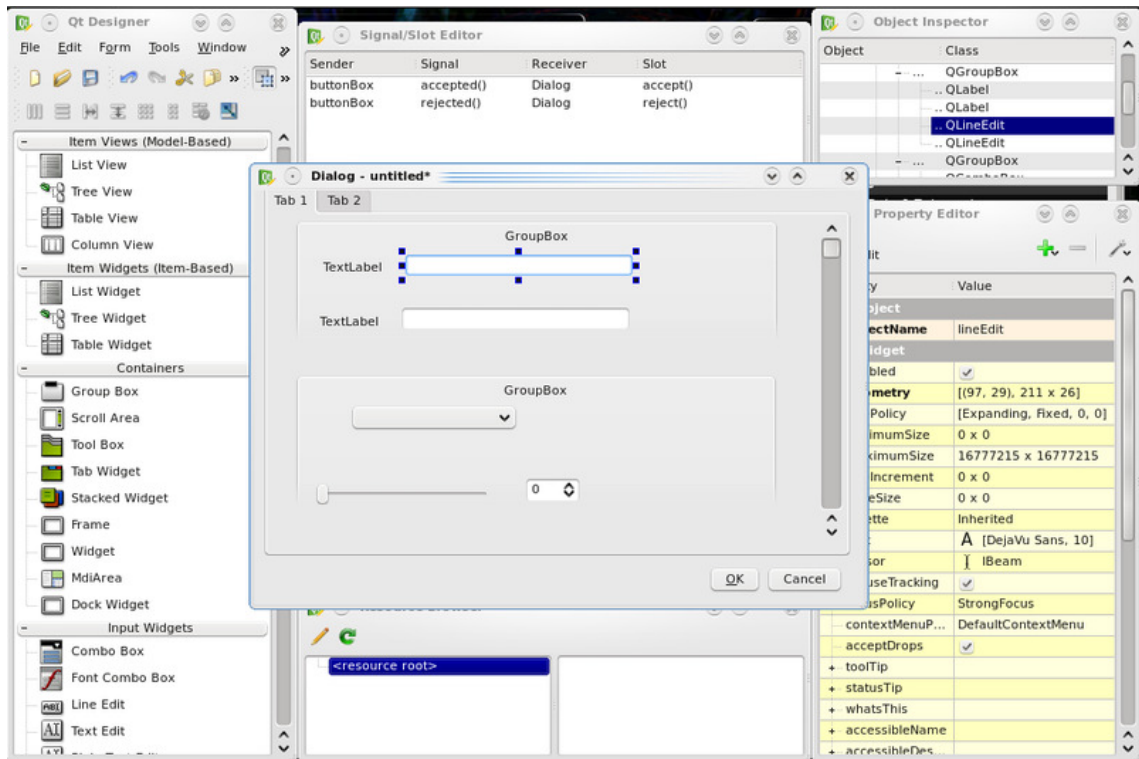


Figura 4-1 Qt Designer

4.1.3 Cálculos Numéricos y Matrices: Librería NumPy

Numpy es una extensión al lenguaje de programación Python, añadiendo soporte para arreglos y matrices multidimensionales de gran tamaño junto con una

extensa librería de funciones matemáticas de alto nivel que operan sobre estos arreglos. NumPy es código abierto.

En la construcción de la herramienta software de este proyecto todos los cálculos son realizados utilizando objetos tipo Array, de la librería Numpy.

El hecho de utilizar estas matrices hace que la cantidad de líneas de código dedicadas a cálculos matemáticos no fuera tan extensa. Como ejemplo se muestra la función para realizar la aproximación mediante el método de los mínimos cuadrados.

```
def minimos(x,y,g=1):  
    A = np.array([[i**k for k in range(g+1)] for i in x])  
    x_aprox = np.dot(np.linalg.inv(np.dot(A.T,A)), np.dot(A.T,y))  
    return x_aprox
```

Figura 4-2 Código completo para aproximación de mínimos cuadrados.

4.1.4 Gráficos Interactivos: Librería matplotlib

Matplotlib es una librería para la generación de gráficos a partir de objetos de la clase Array de la biblioteca NumPy.

Esta librería proporciona una interfaz de programación de aplicaciones o API (del inglés *application programming interface*) orientada a objetos, lo que permite

incrustar gráficas a aplicaciones utilizando bibliotecas de desarrollo de interfaces gráficas como GTK, Qt o wxPython.

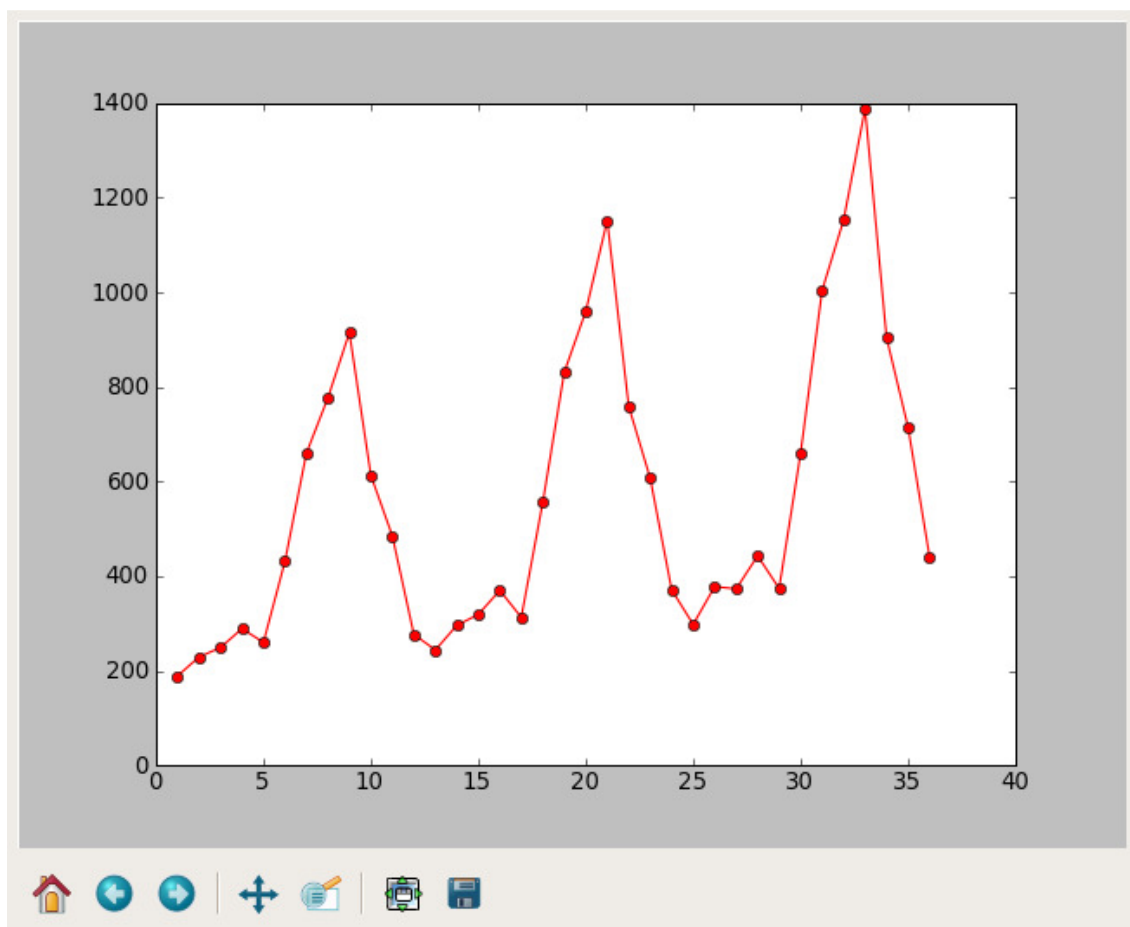


Figura 4-3 Widget de Matplotlib incrustado en la herramienta

En la herramienta se generó una nueva clase a partir de un *widget* genérico proporcionado por la librería matplotlib. Se modificaron los íconos de manipulación para convertirlos a un formato gráfico (PNG) apropiado en todas las plataformas. Y se tradujo la ayuda textual de los íconos y diálogos de configuración.

Este *widget* también permite interactuar con el gráfico generado, haciendo acercamientos, paneos, y navegar entre diferentes vistas. Igualmente se puede regresar a la vista inicial, así como almacenar la vista actual del gráfico en formatos de imagen tales como, PNG y PDF. Formatos vectoriales, SVG y formatos de impresión como Postscript.

4.2 ORGANIZACIÓN DEL CÓDIGO

En el lenguaje Python, los programas están divididos en módulos. Un modulo es un archivo de código fuente, generalmente con la extensión *.py*. La herramienta esta dividida en cinco módulos: *pronostico.py*, *modelos.py*, *ui_pronostico.py*, *estadistica.py* y *mplwidget.py*

4.2.1 Módulo Pronóstico

En este módulo se define la lógica de la interfaz gráfica de la aplicación. El paradigma de programación orientada a eventos es utilizado por la librería Qt, para responder a eventos iniciados por el usuario como oprimir un botón, o modificar el valor de un campo de texto.

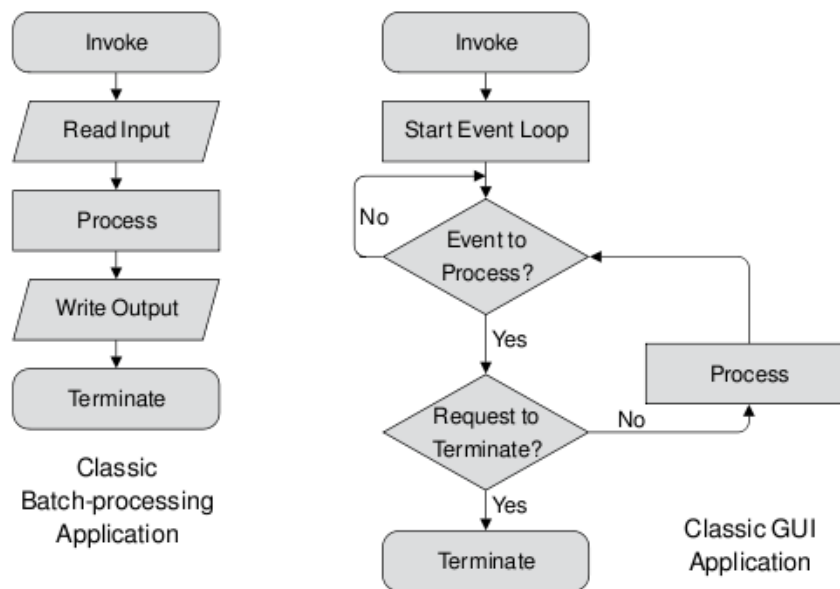


Figura 4-4 Aplicación Clásica por Lotes .vs. Aplicación con Interfaz Gráfica²⁶

Este módulo se apoya en el modulo *ui_pronostico.py* para la configuración automática de algunos elementos de la interfaz de usuario.



Figura 4-5 Objeto QSpinBox para modificar el número de periodos en el año

La parte más importante para el funcionamiento de la aplicación es conectar los eventos con funciones que respondan apropiadamente. Por ejemplo, el objeto *QSpinBox* mostrado en la Figura 4-5 permite al usuario modificar el valor del periodo utilizado en el modelo de pronóstico.

²⁶ Tomada de (Summerfield, 2008)

En el siguiente fragmento de código se muestra como se asocia el evento modificar el valor del periodo, con el método (*change_estacionalidad*), el cual realiza los cálculos y actualiza la interfaz de usuario y el modelo de pronóstico de acuerdo a los nuevos datos.

```
self.connect(self.periodosSpinBox,  
             QtCore.SIGNAL('valueChanged(int)'),  
             self.change_estacionalidad)
```

Figura 4-6 Conectar cambio de periodo con método que lo atienda

El método *change_estacionalidad* verifica que el nuevo valor (variable “*nuevo*”) para el periodo sea válido, es decir que el número de datos en la serie de tiempo sea divisible por el nuevo valor del período. Si el valor es válido, se actualiza el valor del período en el objeto que representa el modelo de pronóstico, y se muestran los cambios en la tabla de datos e interfaz gráfica de usuario.

```
def change_estacionalidad(self, nuevo):  
    if len(self.modelo.t) % nuevo == 0:  
        self.modelo.periodo = nuevo  
        self.update_tablaDatos()  
        if self.estacionalidadCheck.isChecked():  
            self.toggle_estacionalidad(0)  
            self.toggle_estacionalidad(2)  
    else:  
        self.periodosSpinBox.setValue(self.modelo.periodo)
```

Figura 4-7 Método *change_estacionalidad*

4.2.2 Módulo ui_pronostico

Este es un módulo automáticamente generado por la herramienta Qt Designer. Se encarga de la configuración de los elementos de la interfaz gráfica de usuario. En el siguiente fragmento de código se crea la clase para la ventana principal del programa, y en el método setupUI(), se configuran y posicionan los demás elementos de la ventana.

```
class Ui_PronosticoManWindow(object):
    def setupUI(self, puntosMainWindow):
        puntosMainWindow.setObjectName("PronosticoMainWindow")
        puntosMainWindow.resize(800, 600)
        sizePolicy = QtGui.QSizePolicy(QtGui.QSizePolicy.Expanding, QtGui.QSizePolicy.Expanding)
        sizePolicy.setHorizontalStretch(0)
        sizePolicy.setVerticalStretch(0)
        sizePolicy.setHeightForWidth(puntosMainWindow.sizePolicy().hasHeightForWidth())
        puntosMainWindow.setSizePolicy(sizePolicy)
        puntosMainWindow.setBaseSize(QtCore.QSize(550, 220))
        self.centralwidget = QtGui.QWidget(puntosMainWindow)
        self.centralwidget.setObjectName("centralwidget")
        self.verticalLayout_6 = QtGui.QVBoxLayout(self.centralwidget)
        self.verticalLayout_6.setObjectName("verticalLayout_6")
        self.tabWidget = QtGui.QTabWidget(self.centralwidget)
        self.tabWidget.setObjectName("tabWidget")
        self.main = QtGui.QWidget()
        self.main.setObjectName("main")
        self.horizontalLayout_3 = QtGui.QHBoxLayout(self.main)
        self.horizontalLayout_3.setObjectName("horizontalLayout_3")
        self.splitter_3 = QtGui.QSplitter(self.main)
        self.splitter_3.setOrientation(QtCore.Qt.Vertical)
        self.splitter_3.setObjectName("splitter_3")
        self.splitter = QtGui.QSplitter(self.splitter_3)
        self.splitter.setOrientation(QtCore.Qt.Horizontal)
        self.splitter.setObjectName("splitter")
        self.mpl = MplWidget(self.splitter)
```

Figura 4-8 Fragmento modulo ui_pronostico.py

4.2.3 Módulo estadística

En este módulo se ubicaron las funciones relacionadas a cálculos estadísticos que no hacen parte de la clase ModeloMultiplicativo.

```
import numpy as np

def ma(y, n=3):
    w = np.ones(n, 'd')
    return np.convolve(w/w.sum(), y, mode='valid')
```

Figura 4-9 Fragmento Módulo estadística.py

Este módulo utiliza la librería NumPy para trabajar con objetos Array. La función `ma()` mostrada arriba, calcula el promedio móvil de la matriz `y`.

4.2.4 Módulo `mplwidget`

Por último, este módulo contiene el *widget* de `matplotlib`, es decir el objeto que permite visualizar los datos, e interactuar con ellos. Este objeto se puede observar en la Figura 4-3. Además de este objeto, se utiliza un objeto de clase `Figure`, proporcionado por la librería `matplotlib`

En el siguiente código se muestra la creación de esta clase, a partir de la clase `FigureCanvas`.

```
# Matplotlib Figure object
from matplotlib.figure import Figure

class MplCanvas(FigureCanvas):
    """Class to represent the FigureCanvas widget"""
    def __init__(self):
        # setup Matplotlib Figure and Axis
        self.fig = Figure()
        self.ax = self.fig.add_subplot(111)

        # initialization of the canvas
        FigureCanvas.__init__(self, self.fig)
        # we define the widget as expandable
        FigureCanvas.setSizePolicy(self,
                                   QtGui.QSizePolicy.Expanding,
                                   QtGui.QSizePolicy.Expanding)
        # notify the system of updated policy
        FigureCanvas.updateGeometry(self)
```

Figura 4-10 Clase `MplCanvas`

4.3 FUNCIONAMIENTO

Para mostrar y verificar el funcionamiento de la aplicación, se muestra el proceso de resolución del ejercicio tratado en la sección: 2.7 DESCOMPOSICIÓN MULTIPLICATIVA.

4.3.1 Cargar Datos

Se comienza por cargar los datos al programa. Estos datos pueden ser importados desde un archivo externo. Para este ejemplo, se cargan los datos de un archivo en formato *Microsoft Excel*

En la siguiente figura, se muestra la ventana principal del programa ejecutado en el sistema operativo Ubuntu Linux, y el dialogo para cargar los datos de un archivo externo.

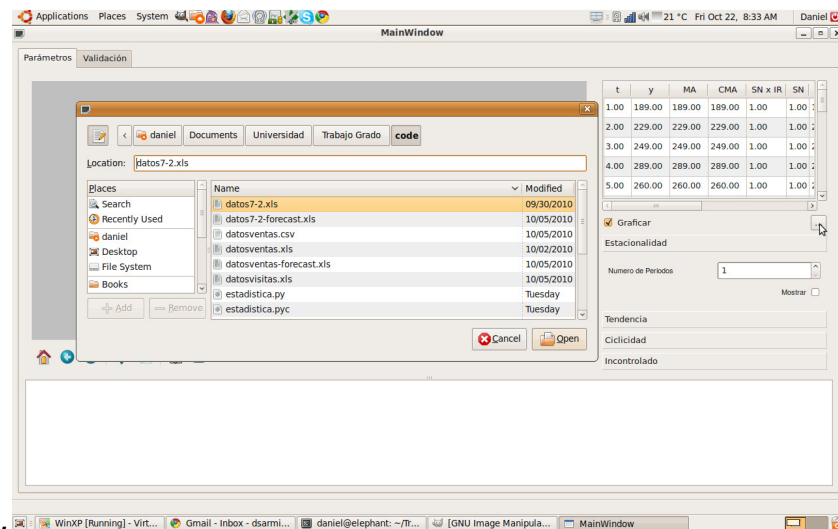


Figura 4-11 Ventana Principal, Dialogo Abrir Archivo

4.3.2 Cálculos

Una vez se han cargado los datos, el programa gráfica los datos de la serie de tiempo, y realiza los cálculos de cada uno de los factores del modelo multiplicativo, en este caso. Los datos son presentados en la tabla de datos principal. La cual se muestra a continuación. Observe que los valores concuerdan con los calculados en el texto guía.

Estos cálculos provienen de la clase `ModeloMultiplicativo`. Cada columna está representada por una propiedad de este objeto. Se muestra el cálculo de los componentes de cada uno de los componentes, y el código que hace esto posible.

t	y	MA	CMA	SN x IR	SN	d	TR	TR x SN	CL x IR	CL	IR
1.00	398.00				1.08	369.82	292.87	315.20	1.26		
2.00	352.00				0.91	386.34	299.18	272.58	1.29	1.24	1.04
3.00	283.00	371.75	371.00	0.76	0.80	353.11	305.48	244.83	1.16	1.22	0.95
4.00	454.00	370.25	369.38	1.23	1.21	374.83	311.78	377.64	1.20	1.17	1.03
5.00	392.00	368.50	367.38	1.07	1.08	364.24	318.09	342.33	1.15	1.17	0.98
6.00	345.00	366.25	358.50	0.96	0.91	378.66	324.39	295.56	1.17	1.12	1.05
7.00	274.00	350.75	338.00	0.81	0.80	341.88	330.69	265.04	1.03	1.05	0.98
8.00	392.00	325.25	308.38	1.27	1.21	323.64	337.00	408.18	0.96	0.93	1.04
9.00	290.00	291.50	284.50	1.02	1.08	269.46	343.30	369.46	0.78	0.80	0.98
10.00	210.00	277.50	276.25	0.76	0.91	230.49	349.60	318.53	0.66	0.74	0.90
11.00	218.00	275.00	286.50	0.76	0.80	272.00	355.91	285.24	0.76	0.76	1.00
12.00	382.00	298.00	314.25	1.22	1.21	315.38	362.21	438.72	0.87	0.87	1.01
13.00	382.00	330.50	340.50	1.12	1.08	354.95	368.51	396.60	0.96	0.94	1.02
14.00	340.00	350.50	359.25	0.95	0.91	373.17	374.82	341.50	1.00	0.98	1.02
15.00	298.00	368.00	373.12	0.80	0.80	371.82	381.12	305.45	0.98	0.98	1.00
16.00	452.00	378.25	382.25	1.18	1.21	373.18	387.42	469.26	0.96	0.98	0.98
17.00	423.00	386.25	391.00	1.08	1.08	393.05	393.73	423.73	1.00	0.99	1.00
18.00	372.00	395.75	397.75	0.94	0.91	408.29	400.03	364.47	1.02	1.02	1.00
19.00	336.00	399.75	395.25	0.85	0.80	419.24	406.33	325.66	1.03	1.00	1.04
20.00	468.00	390.75	382.88	1.22	1.21	386.39	412.64	499.79	0.94	0.94	0.99
21.00	387.00	375.00	366.00	1.06	1.08	359.59	418.94	450.87	0.86	0.86	0.99
22.00	309.00	357.00	348.38	0.89	0.91	339.15	425.24	387.44	0.80	0.81	0.99
23.00	264.00	339.75	342.38	0.77	0.80	329.40	431.55	345.87	0.76	0.77	0.99
24.00	399.00	345.00	355.88	1.12	1.21	329.42	437.85	530.33	0.75	0.79	0.95
25.00	408.00	366.75	382.38	1.07	1.08	379.11	444.15	478.00	0.85	0.86	1.00
26.00	396.00	398.00	423.62	0.93	0.91	434.64	450.46	410.41	0.96	0.96	1.00
27.00	389.00	449.25	470.62	0.83	0.80	485.37	456.76	366.07	1.06	1.03	1.03
28.00	604.00	492.00	506.62	1.19	1.21	498.67	463.06	560.87	1.08	1.10	0.98
29.00	579.00	521.25	536.38	1.08	1.08	538.00	469.37	505.14	1.15	1.14	1.01
30.00	513.00	551.50	558.62	0.92	0.91	563.05	475.67	433.39	1.18	1.22	0.97
31.00	510.00	565.75			0.80	636.34	481.97	386.28	1.32	1.21	1.09
32.00	661.00				1.21	545.73	488.27	591.41	1.12		

Figura 4-12 Cálculos con herramienta software

```

class ModeloMultiplicativo(object):
    def __init__(self, t, y):
        self.t = np.asarray(t)
        self.y = np.asarray(y)
        self.periodo = 1
        self.t_grado = 1
        self.lines = dict()

    def _get_ma(self):
        return estadistica.ma(self.y, self.periodo)
    ma = property(_get_ma)

    def _get_cma(self):
        return self.ma if self.periodo % 2 else estadistica.ma(self.ma, 2)
    cma = property(_get_cma)

    def _get_cma_offset(self):
        return np.ceil(np.median(np.arange(1, self.periodo+1)))
    cma_offset = property(_get_cma_offset)

    def _get_sn_x_ir(self):
        y = self.y
        cma_i = self.cma_offset
        return y[cma_i-1:y.size-cma_i+1] / self.cma
    sn_x_ir = property(_get_sn_x_ir)

    def _get_sn(self):
        periodo = self.periodo
        snt = np.zeros(periodo)
        nc = self.cma.size
        for i in range(periodo):
            subindices = np.arange(i, nc, periodo)
            corrido = (i + self.cma_offset - 1) % periodo
            snt[corrido] = self.sn_x_ir[subindices].sum() / subindices.size
        #normalizar
        snt = (periodo / snt.sum()) * snt
        return np.tile(snt, self.y.size / self.periodo)
    sn = property(_get_sn)

```

Figura 4-13 Clase Modelo Multiplicativo Parte 1

En la figura 4.13, se muestran los atributos correspondiente al promedio móvil (ma), al promedio móvil centrado (cma) y la estacionalidad (sn), los demás atributos y métodos corresponden al constructor de la clase y dos métodos de ayuda: uno para calcular el número de datos que se “pierden” al realizar el

suavizado mediante la media móvil y el otro para calcular la columna estacionalidad por factor incontrolado (sn_x_ir).

Los cálculos realizados en los métodos anteriores son análogos a los cálculos realizados en la resolución del ejercicio mostrado en la sección 2.7

DESCOMPOSICIÓN MULTIPLICATIVA.

Los métodos que se muestran en la figura 4.14 se utilizan para calcular los demás factores del modelo y realizar un pronóstico del siguiente año.

La propiedad d del modelo es el factor estacional compensado, el cual se calcula dividiendo el valor de y por el factor estacional.

Es de anotar que las operaciones se realizan sobre objetos tipo Array, de la librería numpy. Por lo tanto al dividir y en snt en la función d , se están dividiendo todos los elementos de la matriz y (por el elemento de la matriz snt correspondiente).

```

def _get_d(self):
    return self.y / self.sn
d = property(_get_d)

def _get_tr(self):
    return estadistica.poly_eval(self.t, self.tr_poly)
tr = property(_get_tr)

def _get_cl(self):
    return estadistica.ma(self.cl_x_ir, 3)
cl = property(_get_cl)

def _get_ir(self):
    return self.cl_x_ir[1:-1] / self.cl
ir = property(_get_ir)

def _get_tr_poly(self):
    return estadistica.minimos(self.t, self.d, self.t_grado)
tr_poly = property(_get_tr_poly)

def forecast(self):
    periodo = self.periodo
    t_fc = np.arange(self.t[-1], self.t[-1] + 1 + periodo)
    f_sn = np.concatenate([self.sn[-1]], self.sn[:periodo]))
    f_tr = estadistica.poly_eval(t_fc, self.tr_poly)
    return (f_sn * f_tr, t_fc)

```

Figura 4-14 Clase Modelo Multiplicativo. Pt 2.

La propiedad `tr`, la cual corresponde al factor de tendencia del modelo, se calcula realizando una aproximación a un polinomio, cuyo grado es establecido por el usuario, mediante el método de mínimos cuadrados.

Por último, el método *forecast* realiza un pronóstico para el número de períodos utilizados en calcular el factor estacional. Para este pronóstico se calcula el factor de tendencia y el factor de ciclicidad establecidos en el análisis de la serie de tiempo.

4.4 PRUEBAS

Para verificar el funcionamiento de la herramienta y la consecución de los objetivos planteados se realizaron una serie de pruebas para verificar la facilidad de uso de la herramienta y su utilidad para la enseñanza, la habilidad de incorporar nuevo material educativo y la mejora en la comprensión del modelo.

4.4.1 Prueba de Usabilidad

Inicialmente se tenía planeado una sesión presencial de introducción a la herramienta. Una serie de factores como el calendario académico, ya que la parte donde se introdujo la mayor funcionalidad coincidió con un periodo de vacaciones académicas; y que el director pasó a disfrutar de su jubilación, dejando de ser el docente titular de la asignatura. Si bien estos factores se hubieran podido solucionar o mitigar, un análisis de la situación muestra que es más provechoso y acorde con la metodología de desarrollo, adaptarse al cambio y realizar las pruebas sin la introducción presencial.

Esta decisión se justifica en que la mayoría de las aplicaciones de software de escritorio, en la actualidad, se distribuyen mediante enlaces a archivos almacenados en servidores web. En donde se descarga y posteriormente instala la aplicación en el computador personal; rara vez tiene el desarrollador la oportunidad de interactuar de manera presencial con el usuario final. Además, toda aplicación de calidad, y más un software educativo, debe contar con una

ayuda que le permita comprender a un nuevo usuario el funcionamiento de la herramienta.

A pesar de la importancia de la ayuda, una buena herramienta debe contar con una interfaz de usuario intuitiva por lo que inicialmente las pruebas se realizaron sin ofrecer al usuario un manual de ayuda. Aunque esto pueda parecer absurdo, en la práctica, muchos usuarios ya sea por pereza o desconocimiento no leen la documentación y como se mencionó anteriormente, el hecho de no ofrecer ayuda, prueba qué tan intuitiva es la interfaz del software.

Acorde con lo anterior, se puso a disposición de la comunidad educativa mediante la plataforma *moodle* un enlace para descargar un ejecutable de la herramienta. Asimismo un grupo de cinco estudiantes con conocimientos en estadística y que, en semestre anteriores estudiaron el tema de pronóstico, recibieron la herramienta directamente.

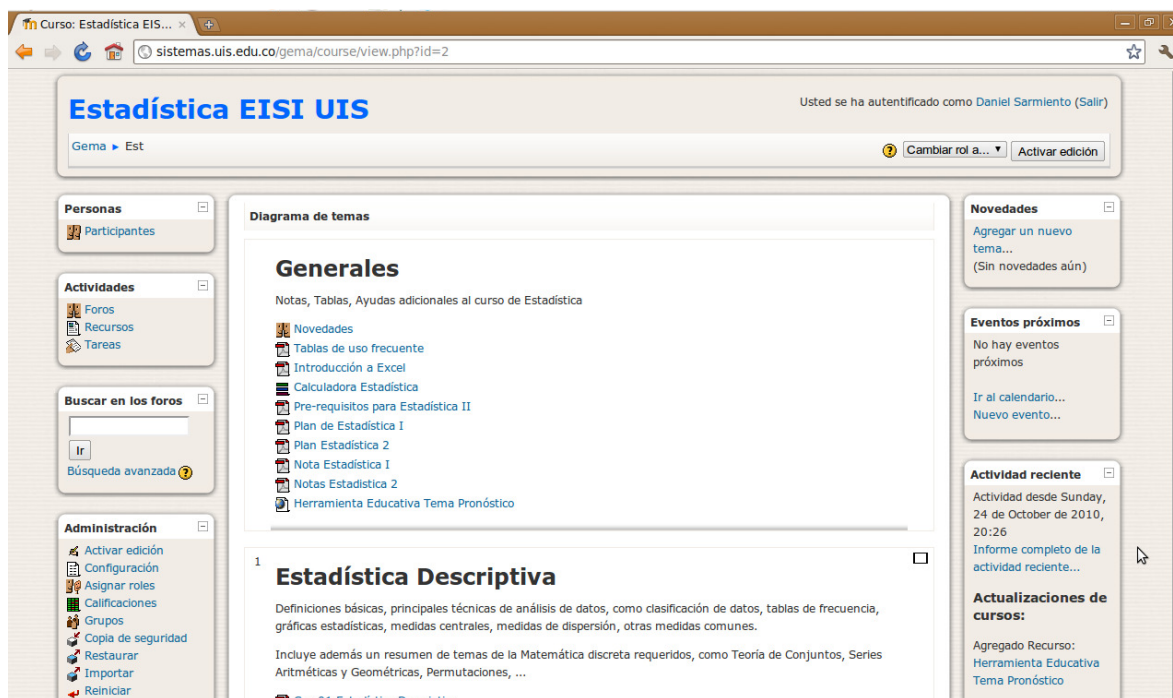


Figura 4-15 Página Principal Curso Estadística Grupo GEMA con enlace a herramienta software para pronóstico.

Los usuarios contaban simplemente con una descripción general del funcionamiento de la herramienta y se asumía cierta familiaridad con el modelo de pronóstico así como conocimientos básicos de estadística (los conocimientos de un estudiante de la asignatura a soportar con la herramienta).

Los resultados de la retroalimentación inicial no fueron muy favorables con respecto a lo intuitivo de la interfaz de usuario. Debido a que las personas que no estaban familiarizadas no entendían la funcionalidad de la herramienta sin una referencia de la temática de pronóstico manejada por la aplicación. Por lo tanto, el hecho de que la mayoría de los usuarios de prueba no estuviera viendo el tema en la asignatura en el momento de la prueba pudo haber influyó. Sin embargo, se decidió incluir un asistente interactivo que se muestra al iniciar el programa y puede ser desactivado cuando el usuario adquiriera familiaridad con la herramienta.

Si bien el asistente es interactivo, a continuación se describen los temas y objetivos del asistente.

- Permitir al usuario familiarizarse con los componentes principales de la interfaz de usuario de la aplicación.
- Guiar al usuario paso a paso a través del desarrollo de un ejercicio de pronóstico utilizando la aplicación.
- Mostrar al usuario información adicional que le permita experimentar con diferentes datos y parámetros de pronóstico.

4.4.2 Prueba de Funcionamiento

Para verificar el funcionamiento de la herramienta en diferentes sistemas operativos, y la veracidad de los cálculos se ingresaron datos con resultados conocidos a la herramienta en diferentes sistemas operativos así como ejecutando el software directamente desde el código y mediante un archivo binario.

Los resultados de ambas pruebas fueron satisfactorios en cuanto a que la aplicación arrojo los mismos resultados, con una tolerancia de tres cifras decimales; a continuación se muestran los cálculos arrojados por la herramienta bajo el sistema operativo Linux (desde el código fuente) y bajo los sistemas operativos Microsoft Windows 7 y Microsoft Windows XP.

Igualmente, los resultados fueron comparados con la literatura. Especialmente los datos en los ejemplos 7.1 y 8.1 del libro de (Bowerman, O'Connel, Koehler; 2007) y ejemplo 12.1 de (Sarmiento, 2010)

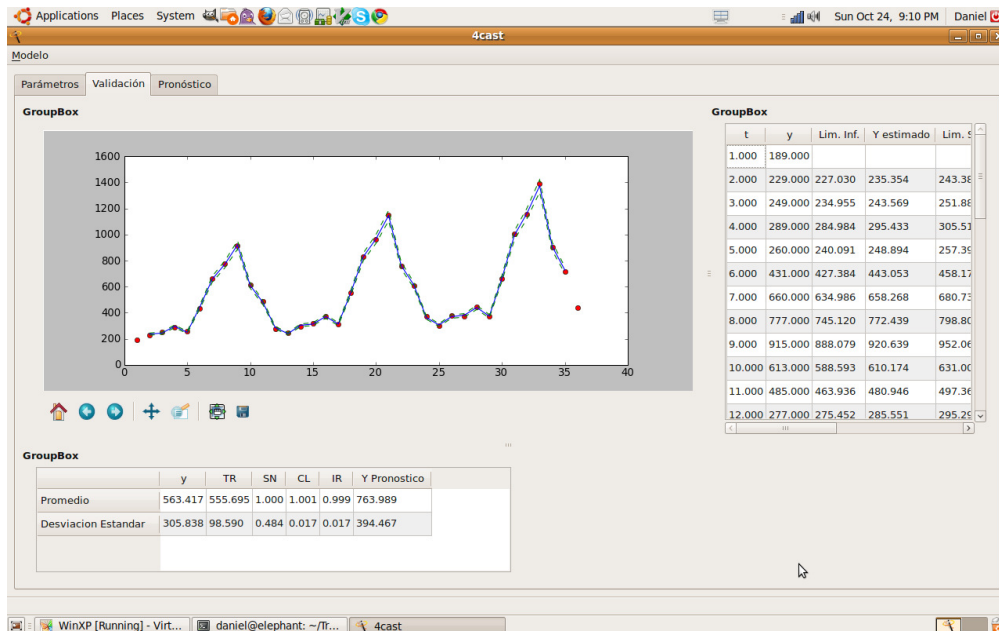


Figura 4-16 Aplicación bajo Linux

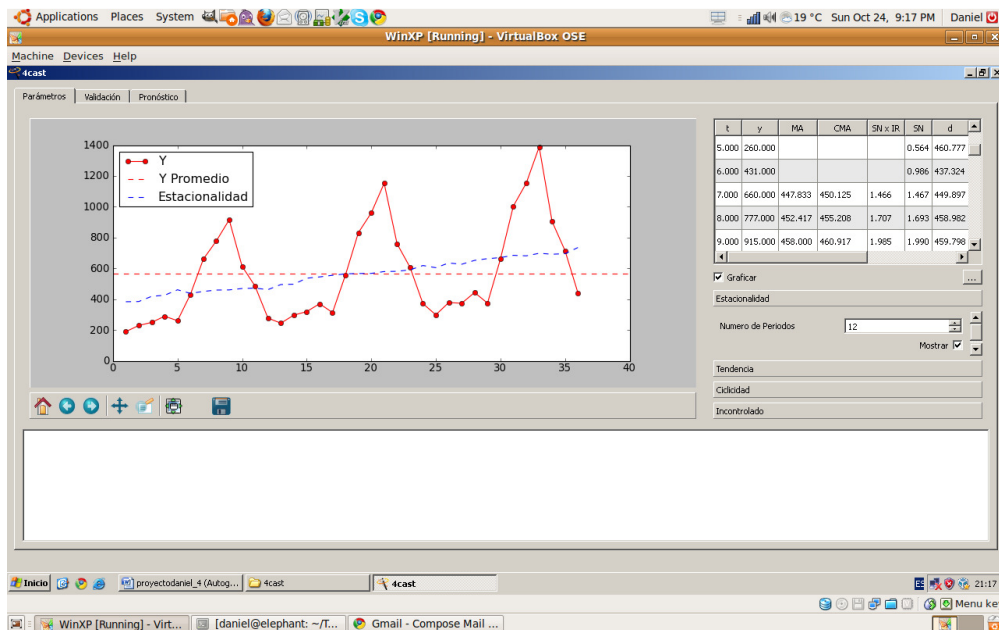


Figura 4-17 Ejecutable Windows Xp (Máquina Virtual)

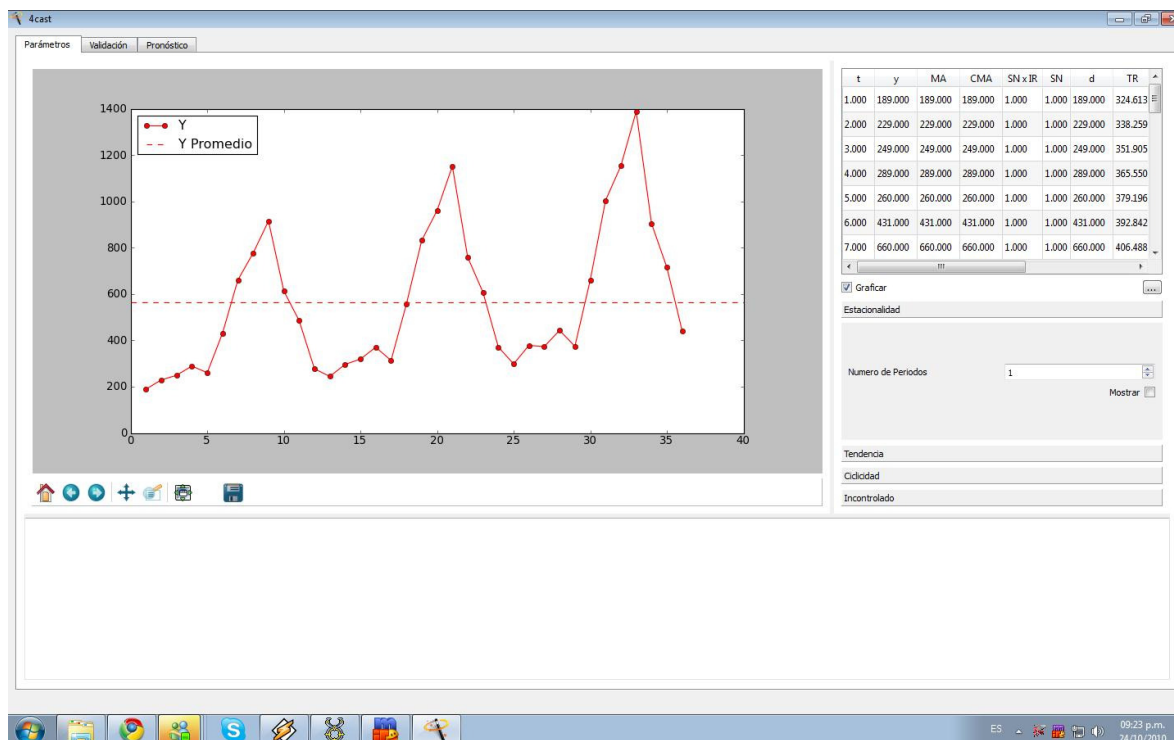


Figura 4-18 Archivo Binario Windows 7

4.4.3 Prueba Nuevo Material

Para obtener datos de situaciones reales y poderlos trabajar con la herramienta, se obtuvieron datos con las visitas a un sitio de internet de música en línea²⁷, y las ventas de un subconjunto de productos de una empresa de manufactura.²⁸

Ambos datos fueron importados de bases de datos relacionales, y almacenados en una hoja de cálculo de Microsoft Excel.

²⁷ Smashits.com

²⁸ Transejes S.A

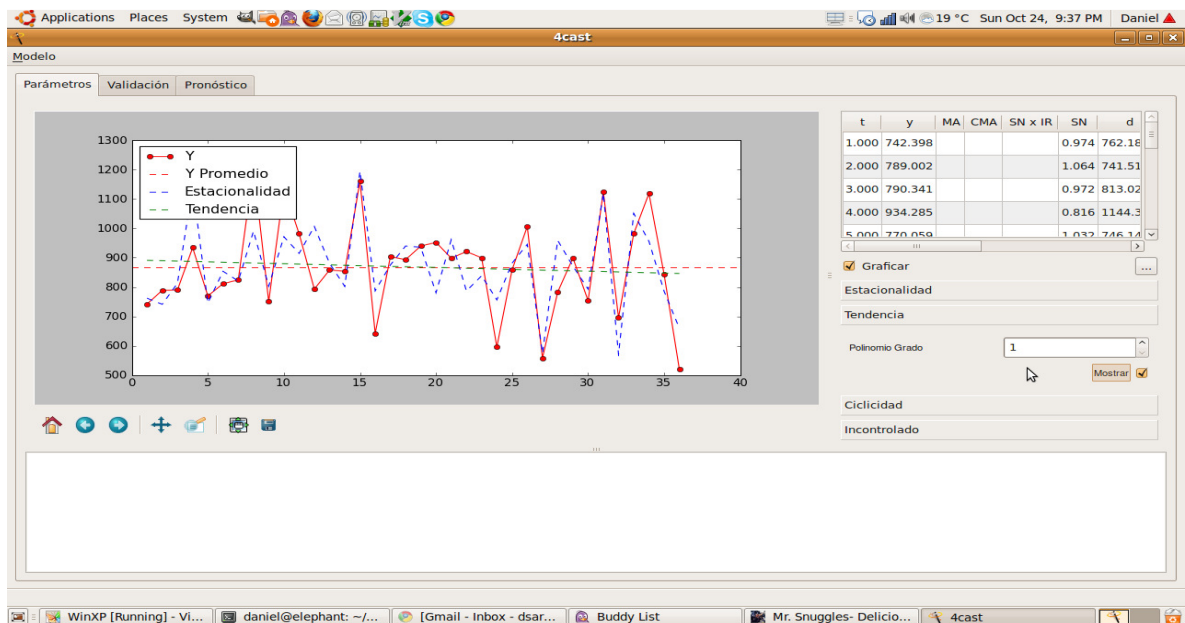


Figura 4-19 Datos ventas, tendencia y estacionalidad

En la figura 4-19 se muestran los datos de las ventas en la pestaña principal de la aplicación. Aquí se han graficado los datos junto con el factor normalizado de la estacionalidad y un polinomio de grado uno, que aproxima el factor de la tendencia.

En la figura 4-20, se presentan los datos de las visitas del sitio de internet Smashits.com. Para estos datos, se utilizó un polinomio cuadrático para aproximar la tendencia, la cual también es graficada junto con el número de visitantes, en millones de personas.

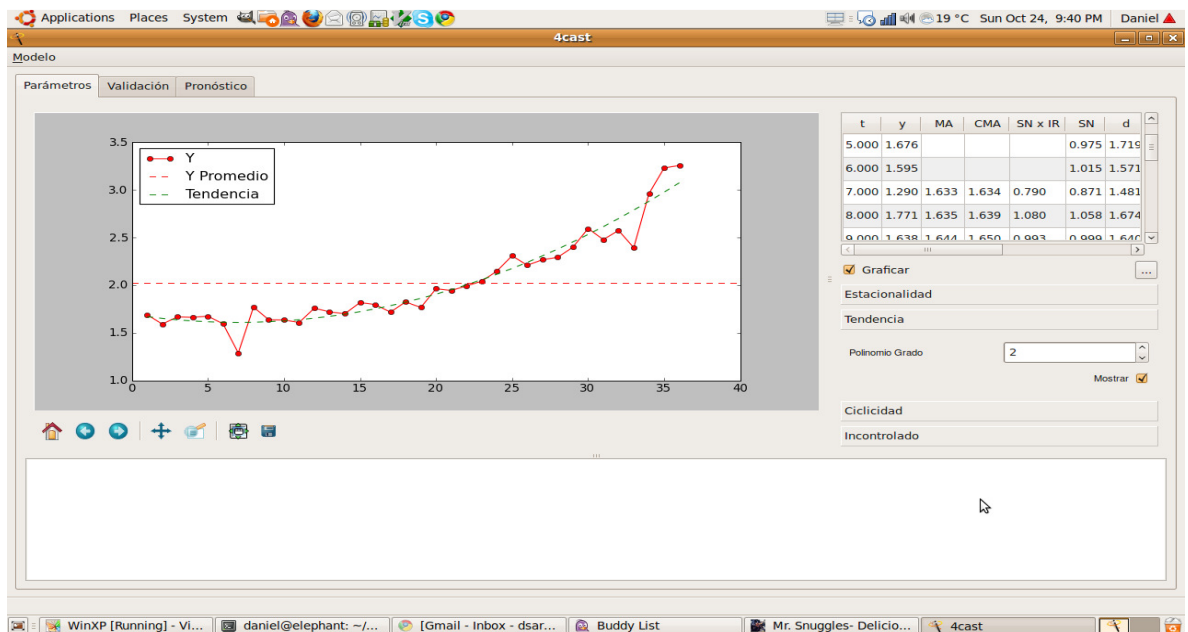


Figura 4-20 Datos visitas Smashits.com y Tendencia (Cuadrática)

Son de especial utilidad los datos de las ventas de la empresa manufacturera ya que se cuenta con los datos de cuatro años.

Se utilizan los tres primeros años para establecer los parámetros del modelo, y realizar un pronóstico del cuarto año. Como se conocen los resultados de este último año, se puede analizar qué tan bueno resultó el modelo y calcular los respectivos indicadores de error y ajustar el modelo.

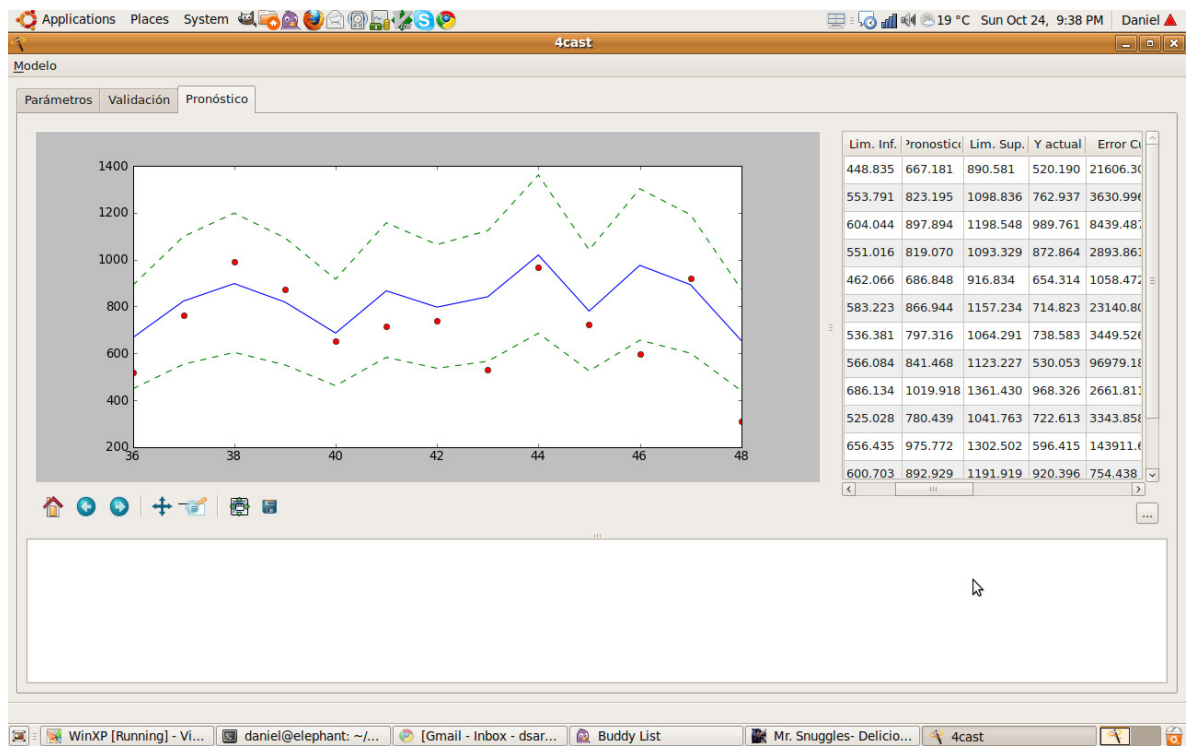


Figura 4-21 Pronóstico e Intervalo de 95% de confianza. Puntos de datos actuales del año pronosticado en rojo

5. CONCLUSIONES

A partir de las metas trazadas inicialmente y el trabajo realizado se derivan una serie de conclusiones y observaciones de interés. Se obtuvo una herramienta para la aplicación de modelos de pronóstico a datos de series de tiempo de acuerdo a los objetivos planteados, al igual que nuevo material educativo y lecciones aprendidas para futuros desarrollos de herramientas similares

5.1 Resultados

Se obtuvo una herramienta software con la cual se pueden importar datos de series de tiempo, establecer parámetros según el modelo de pronóstico para encontrar un patrón en la serie de tiempo, validar los parámetros elegidos y extrapolar el patrón identificado para pronosticar valores futuros.

La principal ventaja de esta herramienta es que está diseñada para trabajar con distintas serie de tiempo estacionales y sin valores faltantes, aunque los resultados serán útiles solo si la serie de tiempo se ajuste a los modelos proporcionados por la herramienta (Ver Marco de Referencia).

Si bien desarrollar una herramienta para una serie de datos específicos hubiera sido más sencillo y quizás hubiera redundado en un programa con una curva de aprendizaje menos pendiente, no se hubieran podido alcanzar los objetivos de aumentar la complejidad y cantidad de material disponible.

Una evidencia de la versatilidad de la herramienta son los dos conjuntos de datos de ejemplo que se obtuvieron durante el desarrollo: los datos de visitas a la página de internet y las ventas de la empresa de manufactura mostrados en la sección 4.4.3.

Estas dos series de tiempo de dominios de aplicación distintos se pudieron trabajar sin problemas en la herramienta. En el caso de los datos de ventas, se contrastaron los pronósticos puntuales para el año siguiente este tipo de validación realizada tomando datos “futuros” reales no era posible con los datos presentados en los ejemplos del curso, antes del desarrollo de la herramienta.

Otro resultado del desarrollo de este software educativo es que los estudiantes tienen a su disposición una herramienta para la generación de gráficos de series de tiempo. El título y los colores utilizados para graficar los datos pueden ser personalizados fácilmente.

Otra ventaja, es que se puede almacenar la imagen en distintos formatos gráficos; esto puede ser utilizado para generar reportes de trabajos de clase.

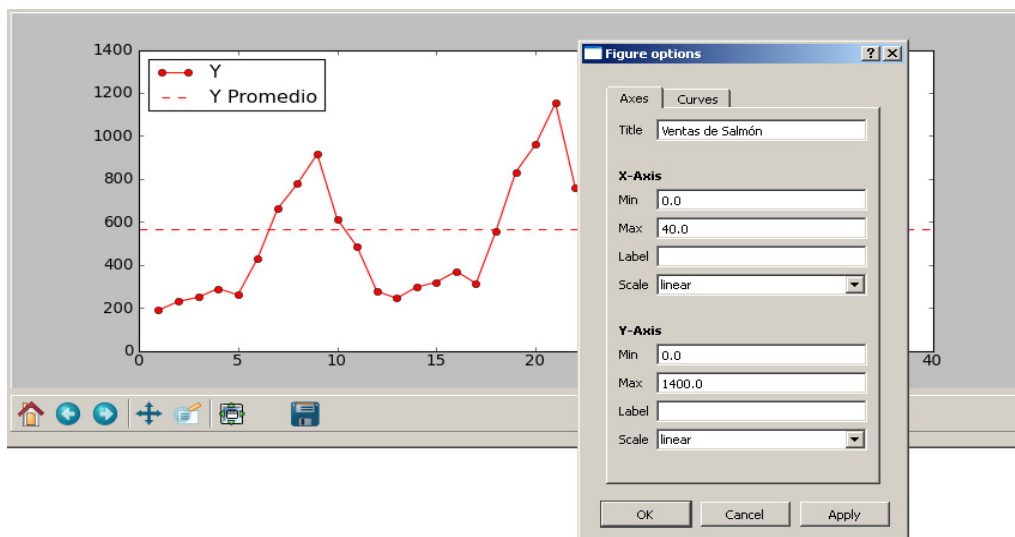


Figura 5-1 Cuadro Modificar Gráfico

Una consecuencia, un tanto inesperada, es que al permitir importar datos externos algunos usuarios manifestaron interés por el análisis de series de tiempo con valores de datos propios. Por ejemplo, un usuario pudo importar los datos que tenía almacenados en una hoja de cálculos con los gastos mensuales de su pequeño negocio.

De esta manera el estudiante logró experimentar con datos de su interés lo que permite practicar con el tema por su propia cuenta y ver la aplicación práctica de la toma de decisiones basadas en pronósticos estadísticos.

Finalmente, el objetivo de poner a disposición de la comunidad universitaria un software que pueda ser modificado y extendido sin ninguna limitación se pudo lograr gracias al uso de código fuente libre. Todas las librerías utilizadas tienen licencias libres, así como el lenguaje de programación utilizado (Python). La aplicación se distribuye siempre acompañada del código fuente.

5.2 Observaciones

Si bien se cumplieron los objetivos y el impacto de la herramienta observado hasta el momento es satisfactorio, se identificaron oportunidades de mejora las cuales se presentan a continuación junto con una serie de observaciones realizadas a lo largo del desarrollo del proyecto.

La principal área de mejora está en la usabilidad de la herramienta. Al entregar una herramienta que permitiera trabajar un amplio conjunto de problemas, se

aumentó la complejidad de la misma lo cual generó confusión en los usuarios al momento de trabajar con la herramienta la primera vez. Como se explicó en el capítulo 4, esto se evidenció al realizar pruebas de usabilidad y tuvo que ser mitigado con un asistente de ayuda que explicara el funcionamiento de la aplicación.

Otro aspecto que puede ser mejorado es el tamaño final del ejecutable de la aplicación. Debido a que Python es un lenguaje interpretado, al generar un ejecutable es necesario incluir el intérprete de Python, así como una serie de librerías que permitan correr el código sin tener dependencias externas. Aunque con la capacidad de almacenamiento actual de los computadores y la velocidad de conexión a la red este es un problema que va perdiendo importancia a medida que pasa el tiempo, este aspecto se debió haber tenido en cuenta desde el inicio del desarrollo de la aplicación.

Por último se pudo haber contado con la participación de la comunidad de usuarios de software libre de la universidad (CUSOL) durante el desarrollo del proyecto. Sin embargo, se hará entrega de una copia de la herramienta terminada junto con su respectivo código fuente a dicho grupo para su divulgación.

BIBLIOGRAFÍA

[3] BOWERMAN, Bruce; O'CONNELL Richard; KOEHLER, Anne. Pronóstico, Series de Tiempo y Regresión Un Enfoque Aplicado. 4ª, Edición, Madrid: Cengage Learning, 2006.

[1] DUNHAM, Penelope H. Calculator Use and Gender Issues. Association for Women in Mathematics Newsletter, 25 (2), 16-18, 995.

[1] PICHLER, Roman. Agile Product Management with Scrum: Creating Products that Customers Love. 1 edition New York: Addison-Wesley, 2010

[3] QUESADA, Antonio R. On the Impact of the First Generation of Graphing Calculators on the Mathematics Curriculum at the Secondary Level. Texto presentado en el panel: *18 Roles of Calculators in the Classroom*. Durante el 8vo congreso de Educación Matemática en Sevilla España. Julio de 1996

[4] SARMIENTO, Enrique. Introducción a la Estadística. 1ª Edición, Bucaramanga: Ediciones UIS, 2010.

[8] SUMMERFIELD, Mark. Rapid GUI Programming with Python and Qt. 1st edition, Michigan: Prentice Hall Editorial, 2008.

[7] WAITS, Berts. The Role of Calculators in Math Education. Texas Instruments Prepared for the Urban Systemic Initiative/Comprehensive Partnership for Mathematics and Science Achievement (USI/CPMSA) Superintendents Forum, 1999

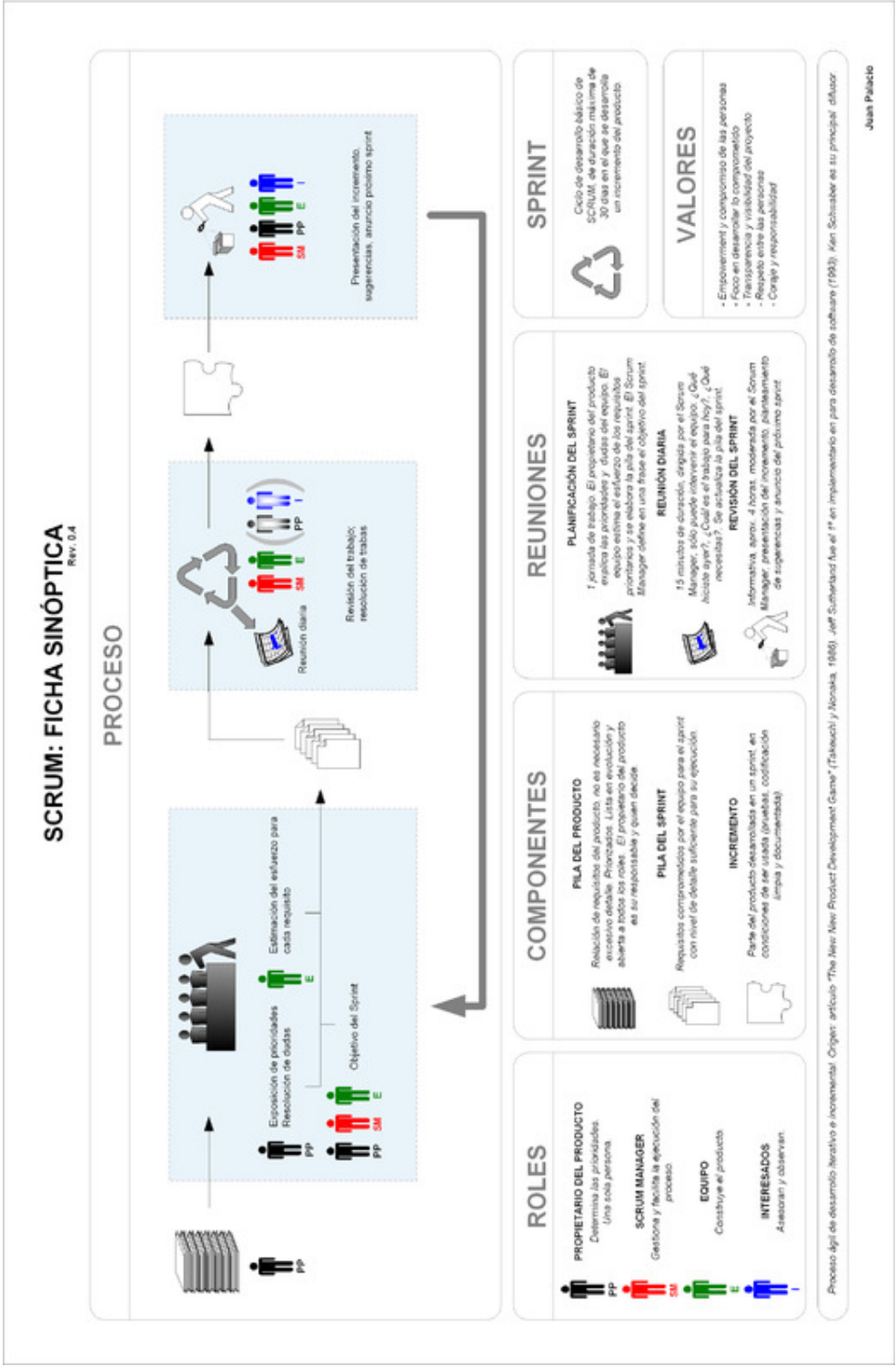
ANEXOS

Se presenta en el Anexo A un resumen del marco de desarrollo SCRUM, el cual fue modificado para establecer el marco de desarrollo propio del proyecto.

En el anexo B se muestra el código fuente de dos módulos.

Los cálculos estadísticos de aproximación de mínimos cuadrados, suavización exponencial y media móvil centrada están agrupados en el módulo estadística. El código completo se incluye de manera ilustrativa y para comprender mejor la implementación del algoritmo presentado en el MARCO DE REFERENCIA.

Anexo A. Ficha Sinóptica Scrum



Anexo B. Código Módulo estadística

```
#vim: set fileencoding=utf-8
```

```
import scipy.stats
```

```
import numpy as np
```

```
def ma(y, n=3):
```

```
    w = np.ones(n, 'd')
```

```
    return np.convolve(w/w.sum(), y, mode='valid')
```

```
def minimos(x,y,g=1):
```

```
    A = np.array([[i**k for k in range(g+1)] for i in  
x])
```

```
    x_aprox = np.dot(np.linalg.inv(np.dot(A.T,A)),  
np.dot(A.T,y))
```

```
    return x_aprox
```

```
def poly_eval(x, coeff):
```

```
    A = np.array([[i**k for k in range(len(coeff))]  
for i in x])
```

```
    return np.dot(A,coeff)
```

```
def intervalo_pronostico_t(x, ir, alfa, gl):
```

```
    tStudent_var_ir = scipy.stats.t.ppf((1-alfa)/2.0,  
gl) * np.std(ir)
```

```
    ir_p = np.average(ir)
```

```
    l_inf = lambda x: x * (ir_p + tStudent_var_ir)
```

```
    l_sup = lambda x: x * (ir_p - tStudent_var_ir)
```

```
    p_inf = np.apply_along_axis(l_inf, 0, x)
```

```
    p_sup = np.apply_along_axis(l_sup, 0, x)
```

```
    return (p_inf, p_sup)
```

```
def ewma(y, alfa):
```

```
    l = np.empty(y.size)
```

```
    umalfa = 1-alfa
```

```
    ini= np.average(y[:y.size/2])
```

```
    l[0] = alfa * y[0] + umalfa * ini
```

```
    for i in range(1, y.size):
```

```
        l[i] = alfa * y[i] + umalfa * l[i-1]
```

```
    return np.concatenate((l[ini], l))
```