# Tarea-Examen 1: Regresión Lineal

Curso Avanzado de Estadística. Profa. Guillermina Eslava Gómez.

Aldo Sayeg Pasos Trejo.

Posgrado en Ciencias Matemáticas. Universidad Nacional Autónoma de México.

21 de febrero de 2020

## 1. Ejercicio 1: Problema 10, capítulo 3 [1]

### 1.1. Inciso a)

Primero visualizamos la tabla de datos

| | Sales | CompPrice | Income | Advertising | Population | Price | ShelveLoc | Age | Education | Urban | US |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 9.50 | 138 | 73 | 11 | 276 | 120 | Bad | 42 | 17 | Yes | Ye |
| 1 | 11.22 | 111 | 48 | 16 | 260 | 83 | Good | 65 | 10 | Yes | Ye |
| 2 | 10.06 | 113 | 35 | 10 | 269 | 80 | Medium | 59 | 12 | Yes | Ye |
| 3 | 7.40 | 117 | 100 | 4 | 466 | 97 | Medium | 55 | 14 | Yes | Ye |
| 4 | 4.15 | 141 | 64 | 3 | 340 | 128 | Bad | 38 | 13 | Yes | No |

Tabla 1: Datos para el ejercicio

Queremos realizar un ajuste lineal para predecir Sales con las variables Price, Urban y US. El código del modelo se puede encontrar en el anexo 2 y su parámetros se muestran en la siguiente tabla

| | Coef. | Std.Err. | t | $P (¿—t—)$ | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| Intercept | 13.043469 | 0.651012 | 20.035674 | 3.626602e-62 | 11.763597 | 14.323341 |
| Urban)[T.Yes] [T.Yes] | -0.021916 | 0.271650 | -0.080678 | 9.357389e-01 | -0.555973 | 0.512141 |
| US)[T.Yes] [T.Yes] | 1.200573 | 0.259042 | 4.634673 | 4.860245e-06 | 0.691304 | 1.709841 |
| Price | -0.054459 | 0.005242 | -10.389232 | 1.609917e-22 | -0.064764 | -0.044154 |

Tabla 2: Parámetros del modelo lineal de la ecuación **??**

### 1.2. Incisos b),c)

El modelo tiene la siguiente forma:

$$\text{Sales} = \beta_0 + \beta_1 \cdot \text{Price} + \beta_2 I_{\text{Urban=NO}}(\text{Urban}) + \beta_3 I_{\text{US=NO}}(\text{US}) \tag{1}$$

Dónde $I_{\text{Urban=NO}}(\text{Urban})$ es una variable indicadora que toma el valor 1 si Urban = Yes y 0 en el caso contrario. De igual manera, $I_{\text{US=NO}}(\text{US})$ toma el valor 1 si US = NO.

Notemos que, de inmediato, podemos interpretar a $\beta_0$ como el valor promedio de Sales cuando US = NO y Urban = NO. $\beta_1$ es la pendiente de Price, es decir, el cambio unitario en Sales por un cambio unitario en Price. $\beta_2$ es una penalización o un cambio del valor promedio de Price para cuando Urban = Yes y, análogamente, $\beta_3$ es otro cambio al valor promedio de Price para cuando US = NO.

### 1.3. Inciso d)

Para hacer la prueba de hipótesis $\beta_i = 0$ para cada coeficiente, podemos fijarnos en el p-value de cada variable en la tabla 2.

Se puede ver de manera clara que la variable Urban tiene un p-value demasiado alto, lo que nos invita a aceptar la hipótesis de que $\beta_2 = 0$. Las demás variables si parecen ser repsentativas debido al bajo valor.

### 1.4. Inciso e)

Al ajustar un modelo ahora usando solo las variables Price y US, los coeficientes se muestran en la tabla **??**

| | Coef. | Std.Err. | t | $P (¿—t—)$ | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| Intercept | 13.030793 | 0.630976 | 20.651794 | 7.001379e-65 | 11.79032 | 14.271265 |
| US)[T.Yes] [T.Yes] | 1.199643 | 0.258461 | 4.641485 | 4.707187e-06 | 0.69152 | 1.707766 |
| Price | -0.054478 | 0.005230 | -10.416123 | 1.272157e-22 | -0.06476 | -0.044195 |

Tabla 3: Parámetros del modelo lineal sin la variable Urban

### 1.5. Inciso f)

Podemos comparar ambos modelos al analizar sus estadísticas, que se muestran en la tabla 4

| | Model | R-squared | AIC | BIC | Log-Likelihood | F-statistic |
|---|---|---|---|---|---|---|
| 0 | Sales ~Price + C(Urban) + C(US) | 0.239275 | 1863.312074 | 1879.277932 | -927.656037 | 41.518772 |
| 1 | Sales ~Price + C(US) | 0.239263 | 1861.318648 | 1873.293042 | -927.659324 | 62.431138 |

Tabla 4: Comaparción de modelos

Notemos que el coeficiente $R^2$ de ambos tiene el mismo valor en tres cifras significativas. Fijándonos solo en ese indicador, podríamos pensar que no hay una mejora sustancial del modelo. Explícitamente, su diferencia tiene el valor de $1.25 \cdot 10^{-5}$

Por otro lado, la estadística $F$ tiene un cambio sustancial, que en su pobabilidad se ve reflejada en un orden de magnitud. Nuevamente, eso podría parecer significativo pero el cambio en los ordenes de magnitud de $\mathbb{P}(F)$ va de $10^{-23}$ a $10^{-24}$, lo cual, tomando en cuenta el error numérico de la aritmética de punto flotante, no es realmente representativo.

En cuanto a los p-values de ambos modelos, notamos que el modelo sin la variable US al menos cuenta con la propiedad de que todos sus p-values son muy bajos.

En resumen, concluyo que no hay una mejora sustancial en el nuevo modelo.

## 1.6. Inciso g)

Los intervalos de confianza para $\alpha = 0.05$ para el modelo sin la variable Urban se pueden encontrar en la tabla 5

| | $L$ | $U$ |
|---|---|---|
| Intercept | 11.790320 | 14.271265 |
| C (US)[T.Yes] | 0.691520 | 1.707766 |
| Price | -0.064760 | -0.044195 |

Tabla 5: Intervalos de confianza para el modelo sin Urban

## 1.7. Inciso h)

Para analizar si existen outliers o puntos de alta influencia, podemos analizar las gráficas de residuales que se muestran a continuación.


Figura 1: Residuales como función del valor ajustado

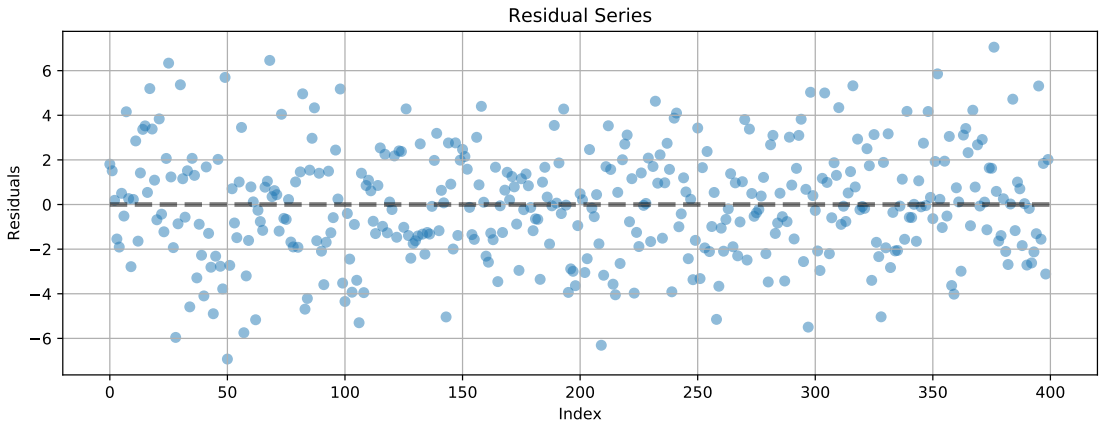


Figura 2: Residuales como función del índice

En principio, aunque hay una gran cantidad de puntos que tienen un residuo grande, no hay ninguo que destaque en particular por tener un residuo demasiado alejado de los otros. Pensando que este es uno de los mejores criterios para encontrar outliers, podemos concluir que en realidad no existen dichos puntos en nuestro conjunto de datos.

Podemos ver más figuras del sistema para confirmar dicho hecho:

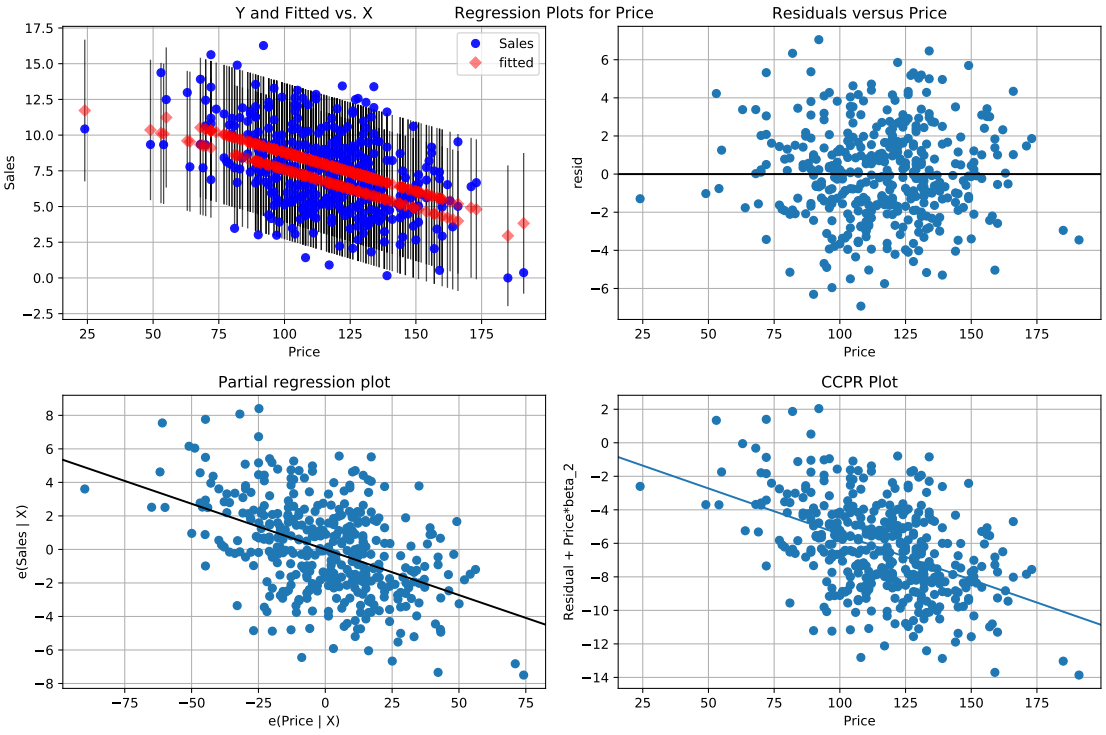


Figura 3: Ajuste y CCPR

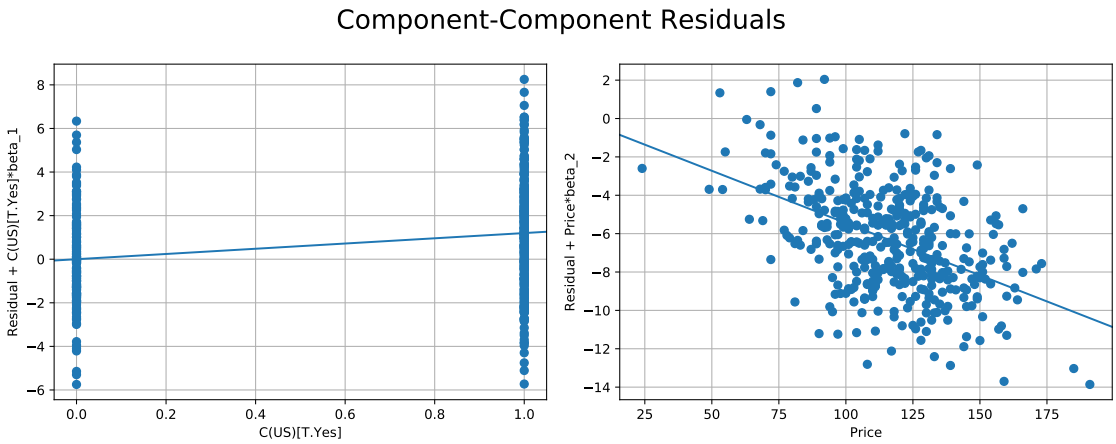


Figura 4: Residuales para cada variable

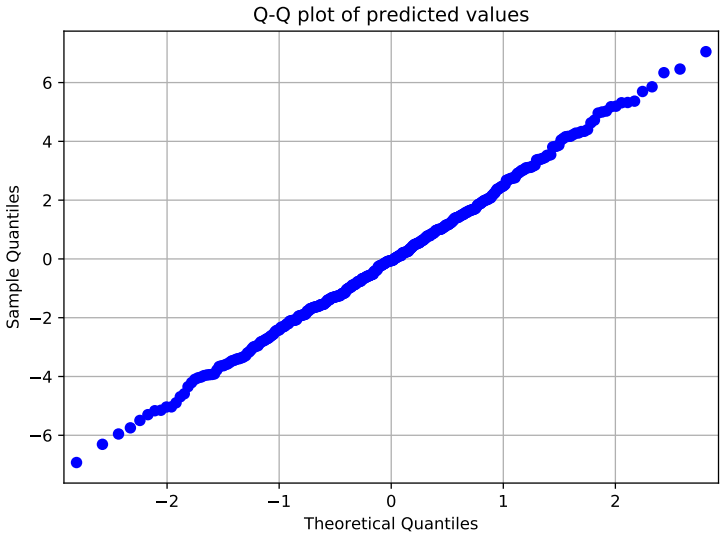


Figura 5: qq-plot para los cuantiles de los residuales

En ninguno de estos puntos detectamos outliers en nuestros datos. Por otro lado, para intentar verificar si existen puntos de alta influencia podemos ver la gráfica del coeficiente $H$ en la figura 6

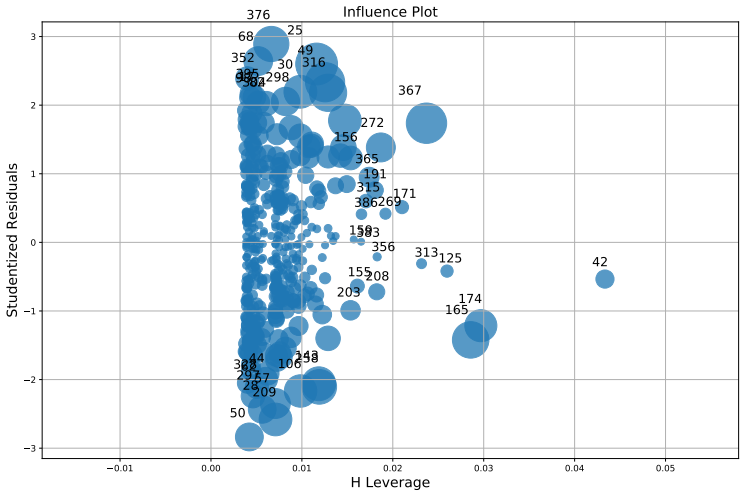


Figura 6: qq-plot para los cuantiles de los residuales

Pensando en la gráfica del coeficiente $H$, aunque tienen un valor pequeño, nos señala a los puntos de datos 42,174 y 165 como posibles puntos de alta influencia. Podemos ver explícitamente sus valores en la base de datos en la tabla 6

| | Sales | CompPrice | Income | Advertising | Population | Price | ShelveLoc | Age | Education | Urban |
|---|---|---|---|---|---|---|---|---|---|---|
| 42 | 10.43 | 77 | 69 | 0 | 25 | 24 | Medium | 50 | 18 | Yes |
| 174 | 0.00 | 139 | 24 | 0 | 358 | 185 | Medium | 79 | 15 | No |
| 165 | 0.37 | 147 | 58 | 7 | 100 | 191 | Bad | 27 | 15 | Yes |

Tabla 6: Posibles puntos de alta influencia

Es claro que todos tiene un valor inusual de Price, pero no afirmaría que ninguno es de alta influencia debido a que su coeficiente $H$ sigue siendo bastante pequeño relativo a los valores esperados para puntos de alto impacto ($H \geq 0.2$) [1].

# 2. Ejercicio 1: Problema 10, capítulo 3 [1]

## 2.1. Inciso a)

Generamos los datos de manera aleatoria como indica el problema y como se muestra en el anexo 2 del reporte, para obtener los datos que muestra en la tabla 7

| | x1 | x2 | y |
|---|---|---|---|
| 0 | 0.417022 | 0.240074 | 2.949735 |
| 1 | 0.720324 | 0.157942 | 3.261717 |
| 2 | 0.000114 | -0.030563 | 3.322517 |
| 3 | 0.302333 | 0.233964 | 2.387546 |
| 4 | 0.146756 | 0.096387 | 3.002498 |

Tabla 7: Posibles puntos de alta influencia

Nosotros sabemos que los datos tienen la forma:

$$\begin{aligned} y &= \beta_0 + \beta_1 \cdot x1 + \beta_2 \cdot x2 + \epsilon \\ &= 2 + 2 \cdot x1 + 0.3 \cdot x2 + \epsilon \\ &= 2 + 2 \cdot x1 + 0.3 \cdot 0.5 \cdot x1 + \epsilon \\ &= 2 + 2.15 \cdot x1 + \epsilon \end{aligned} \tag{2}$$

## 2.2. Inciso b)

La figura 7 muestra un scatter plot entre x1 y x2 así como su correlación

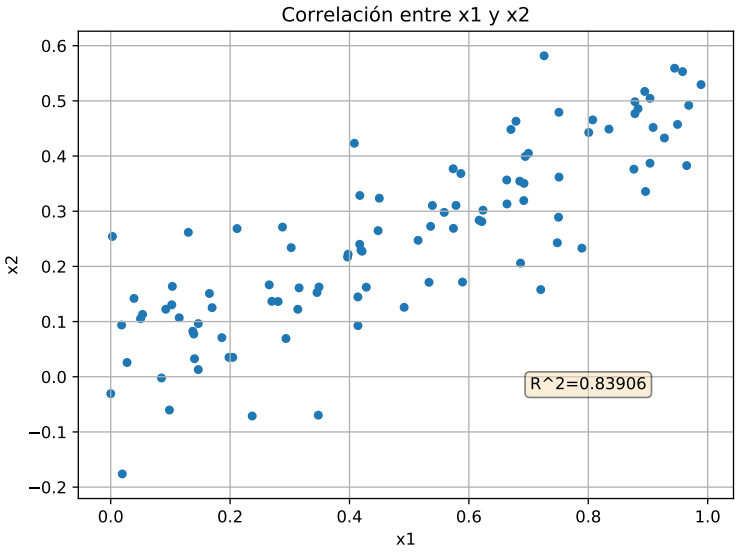


Figura 7: Correlación entre x1 y x2

Aunque, debido a la escala, en la gráfica no se vea de manera muy clara, existe una alta correlación entre ambos puntos, Esto se asenta como un hecho cuando vemos que el valor de $R^2$ es sumamente alto.

## 2.3. Inciso c)

Procedemos a realizar una regresión lineal tomando en cuenta ambas variables, para obtener los coeficientes de la tabla 8

| | Coef. | Std.Err. | t | $P (¿—t—)$ | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| Intercept | 2.189284 | 0.198655 | 11.020556 | 8.532403e-19 | 1.795010 | 2.583559 |
| x1 | 0.704629 | 0.636765 | 1.106575 | 2.712146e-01 | -0.559175 | 1.968432 |
| x2 | 2.502405 | 1.140433 | 2.194259 | 3.060418e-02 | 0.238962 | 4.765848 |

Tabla 8: Coeficientes para el modelo lineal de ambas variables

El valor de los coeficientes obtenidos es sumamente distinto de la ecuación 2. En particular, $\beta_1$ está muy lejos en valor porcentual del verdadero valor. Fijandonos en los p-values, podríamos aceptar

la hipótesis de $\beta_1 = 0$ y quizá también la de $\beta_2 = 0$. Sin embargo,aunque no es suficientemente bajo, el p-value de $\beta_2$ esta un orden de magnitud abajo del de $\beta_1$, lo que complica la desición de aceptar $\beta_2 = 0$.

## 2.4. Inciso d)

Tomando en cuenta solo a x1, obtenemos los coeficientes de la tabla 9

| | Coef. | Std.Err. | t | $P (¿—t—)$ | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| Intercept | 2.248581 | 0.200602 | 11.209167 | 2.942644e-19 | 1.850493 | 2.646669 |
| x1 | 1.876987 | 0.353104 | 5.315681 | 6.683125e-07 | 1.176264 | 2.577709 |

Tabla 9: Coeficientes para el modelo lineal de x1

Es claro que los p-values son sumamente bajos, lo que implica rechazar la hipótesis de que $\beta_1 = 0$.

## 2.5. Inciso e)

Por último, tomando en cuenta solo a x2, obtenemos los coeficientes de la tabla 10

| | Coef. | Std.Err. | t | $P (¿—t—)$ | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| Intercept | 2.265526 | 0.186537 | 12.145167 | 2.952030e-21 | 1.895349 | 2.635703 |
| x2 | 3.561276 | 0.621151 | 5.733353 | 1.090964e-07 | 2.328623 | 4.793930 |

Tabla 10: Coeficientes para el modelo lineal de x2

Nuevamente nos inclinamos a rechazar $\beta_1 = 0$ debido al bajo p-value.

## 2.6. Inciso f)

Para comparar los tres modelos, podemos ver la tabla **??** que compara las estadísticas imporantes de los tres modelos

| Model | R-squared | AIC | BIC | Log-Likelihood | F-statistic | Prob (F-statistic) |
|---|---|---|---|---|---|---|
| y ˜x1 +x2 | 0.260508 | 290.679208 | 298.494719 | -142.339604 | 17.085577 | 4.398146e-07 |
| y ˜x1 | 0.223802 | 293.523632 | 298.733973 | -144.761816 | 28.256462 | 6.683125e-07 |
| y ˜x2 | 0.251173 | 289.933686 | 295.144026 | -142.966843 | 32.871342 | 1.090964e-07 |

Tabla 11: Comparación de modelos

Notemos que ni el $R^2$ ni el logaritmo de la verosimilitud muestran una mejora sustancial entre los modelos, aunque cabe señalar que el de dos variables tiene el mayor valor. El valor de la estadística $F$ para los tres si tiene un valor distinto, pero la prueba $P(> F)$ no cambia sustancialmente, por lo que nuevamente no encontramos diferencia significativa entre los modelos
La respuesta a la pregunta de si estos resultados de los tres incisos anteriores son consistentes entre sí, en principio, pareciera ser que no, que existe una inconsistencia entre ambos.

Sin embargo, retomando el hecho de que los puntos son colineales y fueron generados añadiendo números aleatorios con distribución normal, se puede explicar que la alta correlación entre sus variables hace más inestable la regresión y hace difícil de comparar la aceptación u rechazo de hipótesis entre los tres casos.

## 2.7. Inciso g)

Si añadimos una observación $(x1, x2, y) = (0.1, 0.8, 0.6)$, podemos primero visualizar el valor en el plano $x1 - x2$, mostrado en la figura 8, para entender si será un punto de alta influencia.

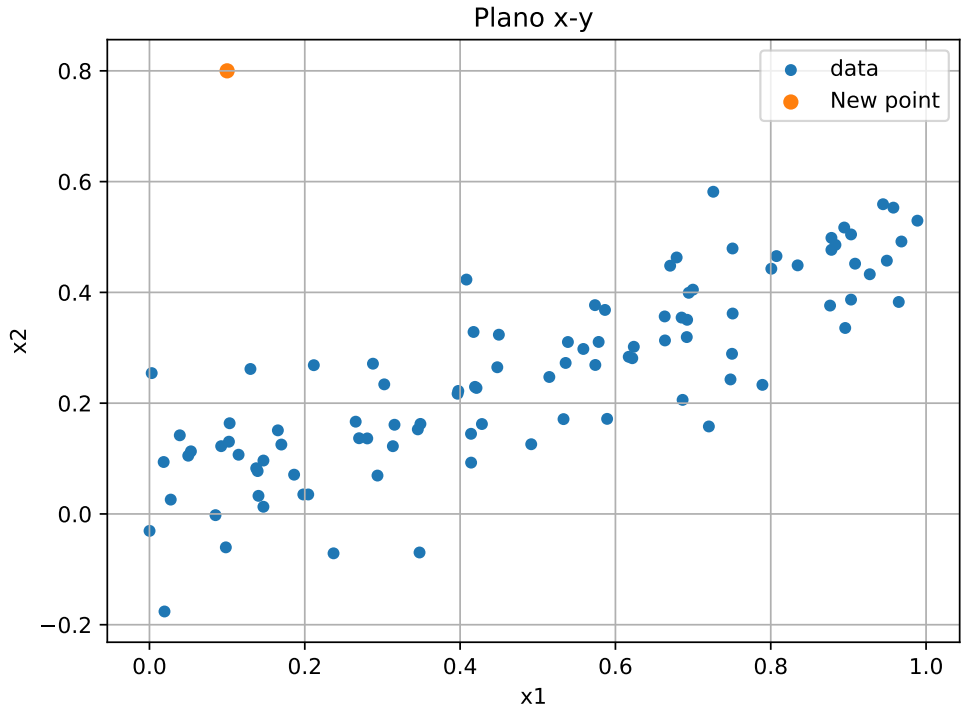


Figura 8: Nueva observación

Parece ser un dato muy alejado de los valores normales de las otras observaciones, por lo que podría considerarse de alta influencia. Para ver si también es un outlier, podemos realizar los mismos ajustes anteriores:

| | Coef. | Std.Err. | t | $P (¿—t—)$ | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| Intercept | 2.167608 | 0.205090 | 10.569045 | 7.071833e-18 | 1.760613 | 2.574603 |
| x1 | 1.725411 | 0.535528 | 3.221884 | 1.729105e-03 | 0.662672 | 2.788149 |
| x2 | 0.529136 | 0.917938 | 0.576440 | 5.656396e-01 | -1.292483 | 2.350755 |

Tabla 12: Coeficientes para el nuevo modelo lineal de ambas variables

| | Coef. | Std.Err. | t | $P (¿—t—)$ | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| Intercept | 2.192032 | 0.199988 | 10.960836 | 8.877601e-19 | 1.795213 | 2.588851 |
| x1 | 1.956577 | 0.353723 | 5.531375 | 2.595173e-07 | 1.254713 | 2.658441 |

Tabla 13: Coeficientes para el nuevo modelo lineal de x1

| | Coef. | Std.Err. | t | $P (¿—t—)$ | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| Intercept | 2.430716 | 0.196842 | 12.348559 | 9.275744e-22 | 2.040139 | 2.821293 |
| x2 | 2.743821 | 0.636535 | 4.310557 | 3.849638e-05 | 1.480797 | 4.006845 |

Tabla 14: Coeficientes para el nuevo modelo lineal de x2

Analizamos ahora el residual estandarizado del nuevo punto en todos los modelos:

| Modelo | Residual estandarizado |
|---|---|
| $y \sim x1 + x2$ | -20.8429 |
| $y \sim x1$ | -17.1936 |
| $y \sim x2$ | -36.8806 |

Tabla 15: Residual estandarizado

El valor de su residual estandarizado es muy alto para todos los ajustes, cosa que nos indica también que el punto parece ser un outlier. De ambos valores, podemos concluir que el punto es tanto un outlier como un punto de alta influencia: sus valores tanto en $y$ como en $x1$,$x2$ son inusuales para los del conjunto de datos.

En general, también observamos que $R^2$ y la estadística $F$ no mejoraron mucho para estos ajustes en comparación con los anteriores. La diferencia más interesante entre estos ajustes y los realizados anteriormente se encuentra en los p-values del ajuste de las dos variables, pues aquí ya no nos permite rechazar la hipótesis de que $\beta_1 = 0$ de manera tan sencilla.

Los coeficientes del modelo de dos variables también mejoran, aunque no sustancialemente, para asemejarse al modelo original.

## 3. Ejercicio 2

Tenemos 23137 observaciones de 14 variables, de las cuales queremos usar 13, todas categóricas, para estimar 1 llamada “Crash_Score”. Los datos se pueden observar en la tabla 17

| | Crash_Score | year | Month | Time_of_Day | Rd_Feature | Rd_Character | Rd_Class | Rd_Configuration | Rd_S |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 6.56 | 2016 | 6 | 2 | NONE | STRAIGHT-LEVEL | STATE HWY | TWO-WAY-PROTECTED-MEDIAN | SMO ASPI |
| 1 | 6.53 | 2016 | 6 | 3 | NONE | STRAIGHT-LEVEL | OTHER | TWO-WAY-NO-MEDIAN | COA ASPI |
| 2 | 1.58 | 2016 | 6 | 5 | NONE | STRAIGHT-LEVEL | STATE HWY | TWO-WAY-NO-MEDIAN | SMO ASPI |
| 3 | 7.15 | 2016 | 6 | 3 | NONE | STRAIGHT-LEVEL | OTHER | TWO-WAY-NO-MEDIAN | SMO ASPI |
| 4 | 9.57 | 2016 | 6 | 6 | NONE | STRAIGHT-LEVEL | OTHER | TWO-WAY-NO-MEDIAN | COA ASPI |

Tabla 16: Conjunto de datos

Queremos encontrar un modelo lineal que ajuste de manera buena los datos. Antes de querer tomar un modelo arbitrario, al estar tratando con variables categóricas, debemos tomar alguna como referencia para cada categoría.

Ya que en principio las distribución del Crash_Score puede cambiar entre categorías de la misma variable, realizamos boxplots e histogramas de el valor del Crash_Score para cada categoría. A continuación mostramos dos gráficas para categorías distintas en las figuras9 y 10

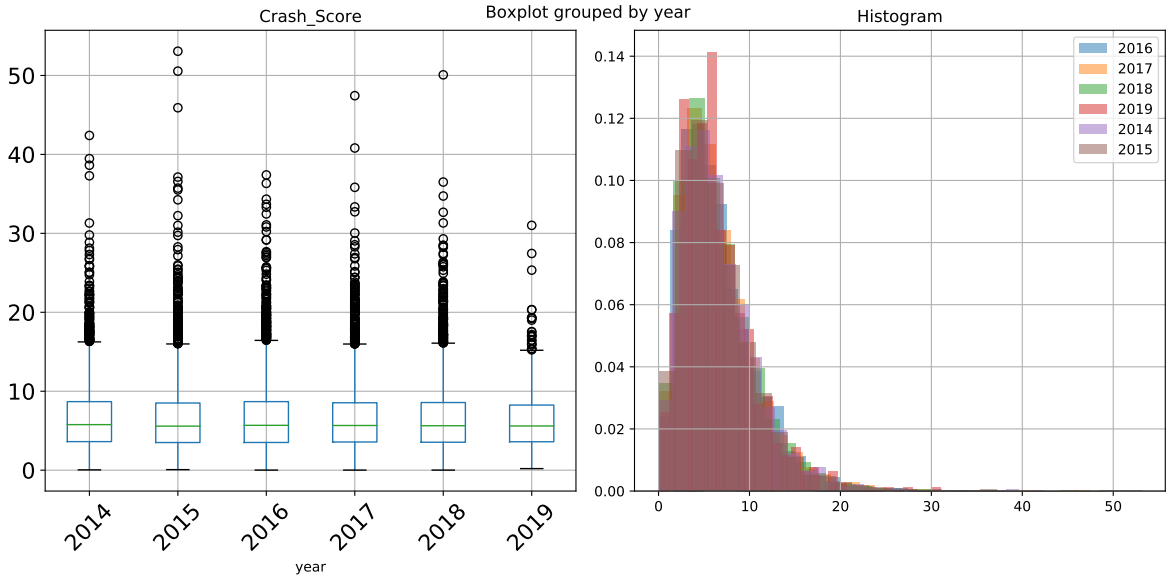


Figura 9: Boxplot e histograma para variable year

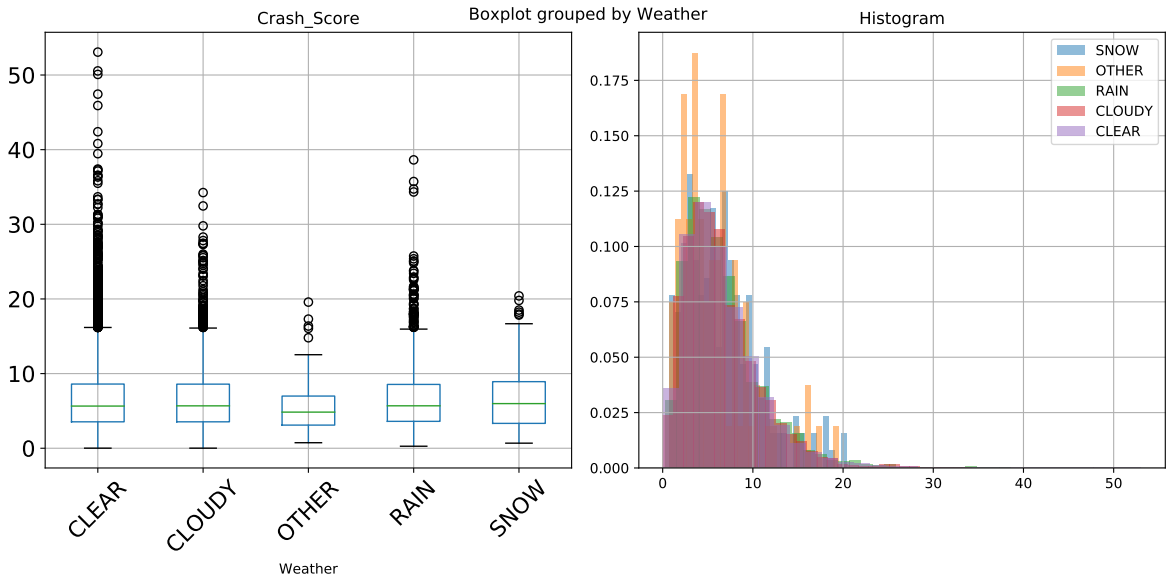


Figura 10: Boxplot e histograma para variable Weather

Antes que cualquier cosa, la forma del boxplot nos indica que la distribución de la variable no es simétrica. Esto provoca que el análisis inmediato de dichas gráficas no sea tan sencillo. En la figura 9 se ve claramente que la distribución por año no tiene ninguna diferencia observable, mientras que en la figura 10 la única distribución que destaca es la de la categoría "OTHER", lo cual se explica en el hecho de que ahí puede haber observaciones o bien mal catalogadas o difíciles de predecir.

Para ver si existe alguna influencia del tiempo, podemos ver el valor del Crash_Score como función del tiempo para una variable de tiempo en unidades arbitrarias definida como $(\text{year} - 2014) \cdot 12 \cdot 6 + \text{Month} \cdot 6 + \text{Time\_of\_Day}$. La figura 11 muestra dicha gráfica.

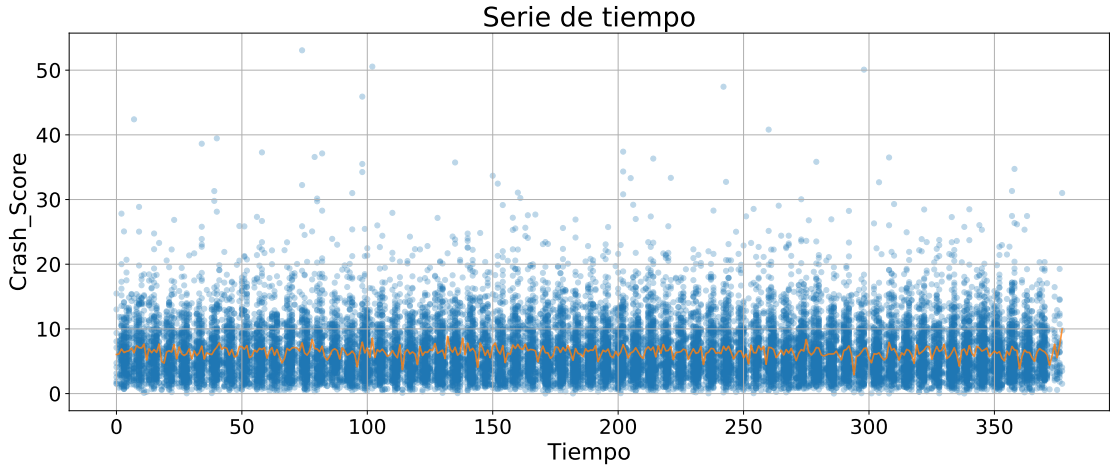


Figura 11: Crash_Score como función del tiempo

Es claro que no parece haber ninguna relación simple entre el tiempo y el valor del Crash_Score. Así, dado que estas observaciones no permiten escoger objetivamente categorías de referencia, se tuvieron que escoger con justificaciones arbitrarias o de sentido común. La tabla **??** muestra la

categoría de referencia para todas las variables

| Variable | Referencia | Justificación |
|---|---|---|
| year | 2014 | Arbitrario |
| Month | 10 | Mes sin vacaciones escolares o personales |
| Time_of_Day | 4 | Horario con hora de comida y no sobrecargado |
| Rd_Feature | NONE | Comparar con caminos normales |
| Rd_Character | STRAIGHT-LEVEL | Compara con caminos sin peralte o curvas |
| Rd_Class | OTHER | Arbitrario |
| Rd_Configuration | TWO-WAY-UNPROTECTED-MEDIAN | Camino más representativo |
| Rd_Surface | SMOOTH ASPHALT | Material más representativo |
| Rd_Conditions | OTHER | Arbitrario |
| Light | DAYLIGHT | Luz más estándar |
| Weather | CLEAR | Clima más estándar |
| Traffic_Control | NONE | Arbitrario |
| Work_Area | NO | Quitar influencia por tráfico |

Tabla 17: Referencias para cada variable categórica

Se ajustaron en total 7 modelos a los datos, cuyas ecuaciones mostramos a continuación

| Modelo | Ecuación |
|---|---|
| 0 | Crash_Score + year + Month + Time_of_Day + Rd_Feature + Rd_Character + Rd_Class + Rd_Configuration + Rd_Surface T + Rd_Conditions + Light + Weather + Traffic_Control + Work_Area |
| 1 | Crash_Score_boxcox_0_27 + year + Month + Time_of_Day + Rd_Feature + Rd_Character + Rd_Class + Rd_Configuration + Rd_Surface T + Rd_Conditions + Light + Weather + Traffic_Control + Work_Area |
| 2 | Crash_Score_boxcox_0_27 + Time_of_Day + Rd_Feature + Rd_Character + Rd_Class + Rd_Surface T + Light + Traffic_Control |
| 3 | Crash_Score_boxcox_0_27 + Rd_Class + Traffic_Control + Time_of_Day*Light + Rd_Feature*Rd_Character*Rd_Surface |
| 4 | Crash_Score_boxcox_0_27 + Rd_Class + Traffic_Control + Rd_Feature*Rd_Character*Rd_Surface*Time_of_Day + Rd_Feature*Rd_Character*Rd_Surface*Light |
| 5 | Crash_Score_boxcox_0_27 + Rd_Class + Traffic_Control + Rd_Feature+ Rd_Character+ Rd_Surface+ Time_of_Day+ Light+ Rd_Feature:Rd_Character+ Rd_Feature:Rd_Surface+ Rd_Feature:Time_of_Day+ Rd_Character:Time_of_Day+ Rd_Surface:Time_of_Day+ Rd_Feature:Light+ Rd_Character:Light+ Rd_Feature:Rd_Character:Rd_Surface+ Rd_Feature:Rd_Surface:Time_of_Day+ Rd_Feature:Rd_Character:Light+ Rd_Feature:Rd_Character:Rd_Surface:Light |
| 6 | Crash_Score_boxcox_0_27 + Rd_Class + Traffic_Control + Rd_Feature+ Rd_Character+ Rd_Surface+ Time_of_Day+ Light+ Rd_Feature:Rd_Character+ Rd_Feature:Rd_Surface+ Rd_Surface:Time_of_Day+ Rd_Character:Light+ Rd_Feature:Rd_Surface:Time_of_Day+ Rd_Feature:Rd_Character:Rd_Surface:Light |

Tabla 18: Ecuaciones de cada modelo

La variable "Crash_Score_boxcox_0_27" representa a "Crash_Score" después de haber realizado un transformación Box-Cox con $\lambda = 0.27$. Se tomó ese valor de lambda pues ese era el que maximizaba el logaritmo de la verosimilitud, como se muestra en la figura 12

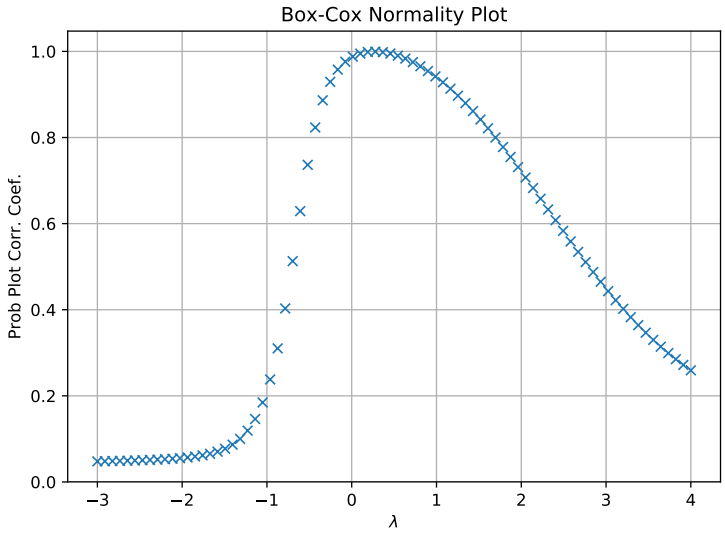


Figura 12: verosimilitud como función de $\lambda$ para transformaciones boxcox de la variable Crash_Score

Empezamos en el modelo 0, un modelo lineal normal, y luego pasamos al modelo 1 haciendo la transformación boxcox. Para pasar al modelo 2, buscábamos reducir las variables.

Para reducir variables, se utilizó también un script de R que, utilizando la función stepstats, reducía las variables de modelo hasta obtener un conjunto más pequeño fijándose en el criterio AIC. Las variables que quedaron en el modelo fueron Time_of_Day, Rd_Feature , Rd_Character , Rd_Class , Rd_Surface T, Light y Traffic_Control. Este resultado también era consistente con el análisis ANOVA de las variables para el modelo 1. El modelo 2 trabajaba exactamente con estas variables de manera lineal.

Los modelos 3-6 trabajan con las mismas variables pero realizando distintas interacciones. Las interacciones fueron definidas de manera arbitraria, aunque se buscó que interactuaran entre ellas variables que no tiene mucha relación en el plano semántico (Time_of_Day con las características del camino, por ejemplo)

Para analizar a fondo la representabilidad de las variables de cada modelo, remitimos al lector al anexo 1 del presente trabajo donde se muestran las tablas ANOVA de cada modelo. No se pueden presentar las tablas de los p-values e intervalos de confianza de todas las variables ya que, al codificarse en variables dummies, el número de variables crece considerablemente y no es posible incluir esa información en el presente trabajo

Finalmente, presentamos la siguiente tabla comparando todos los modelos realizados

| model | R-squared | AIC | BIC | Log-Likelihood | F-statistic | Prob (F-statistic) |
|---|---|---|---|---|---|---|
| 0 | 0.017116 | 132646.376793 | 133121.278906 | -66264.188397 | 6.929017 | 2.266000e-52 |
| 1 | 0.019800 | 67662.540049 | 68137.442161 | -33772.270024 | 8.037371 | 2.602486e-64 |
| 2 | 0.018763 | 67630.995836 | 67880.520674 | -33784.497918 | 14.727664 | 5.746884e-74 |
| 3 | 0.026283 | 67707.006692 | 68978.778451 | -33695.503346 | 3.950638 | 2.687955e-55 |
| 4 | 0.048653 | 68165.253070 | 73445.520625 | -33426.626535 | 1.755272 | 1.614963e-28 |
| 5 | 0.040874 | 67983.678981 | 71774.846693 | -33520.839491 | 2.055153 | 2.346331e-35 |
| 6 | 0.039879 | 67953.659358 | 71527.498984 | -33532.829679 | 2.127681 | 4.934881e-37 |

Tabla 19: Comparación de modelos

Fijándo la atención en el parámetro $R^2$, es claro que el modelo 4 presenta el valor más grande de este parámetro. Para $P(> F)$, dicho modelo no presenta el mejor valor. En cuanto al logaritmo de la verosimilitud, es claro que el modelo 4 también presenta el máximo valor de todos los otros modelos. Revisando su qqplot en la figura 13, confirmamos su adecuación.

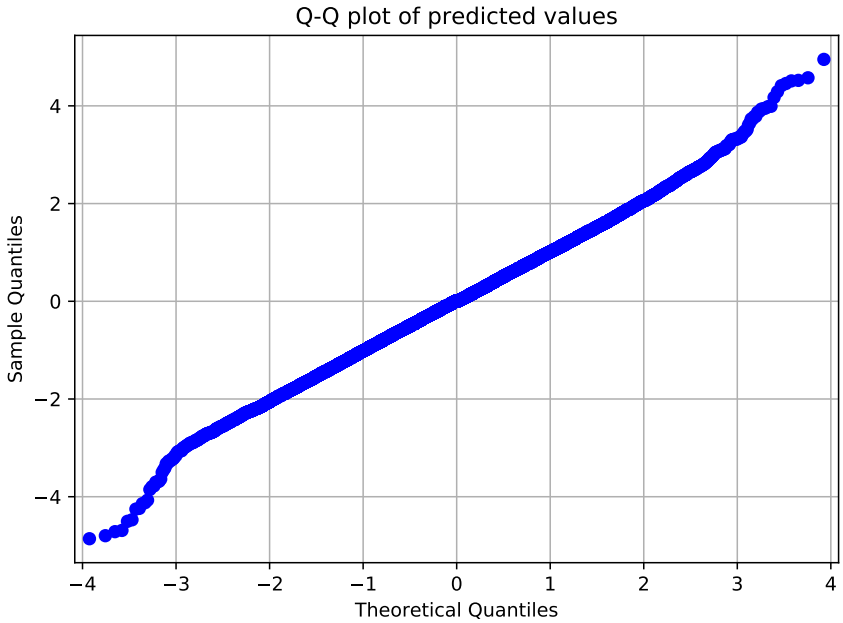


Figura 13: Q-Q Plot de los residuales del modelo 4

Así, podemos concluir que el modelo 4 presenta el mejor ajuste y, tomando en cuenta el valor $R^2$, es el que presenta mayor estabilidad para predicción. Lo distintivo del model es que toma en cuenta las interacciones de la luz del día y la hora con las características del camino. Podemos interpretar que estas interacciones son las más significativas a la hora de predecir cuando puede darse un accidente de tráfico: cuando las características climáticas y la visibilidad, representadas por la luz del día, al igual que el tráfico, representado por la hora, interactúan con las características del camino.

# Anexo 1: Tablas Anova y de Coeficientes de los modelos del ejercicio 2

## Modelo 0

| Variable | df | sum_sq | mean_sq | F | $P (¿F$) |
|---|---|---|---|---|---|
| year | 5.0000 | 61.5800 | 12.3160 | 0.6827 | 0.6366 |
| Month | 11.0000 | 101.7591 | 9.2508 | 0.5128 | 0.8962 |
| Time_of_Day | 5.0000 | 751.2592 | 150.2518 | 8.3282 | 0.0000 |
| Rd_Feature | 4.0000 | 2485.7422 | 621.4356 | 34.4453 | 0.0000 |
| Rd_Character | 6.0000 | 292.0400 | 48.6733 | 2.6979 | 0.0128 |
| Rd_Class | 2.0000 | 2045.5863 | 1022.7931 | 56.6919 | 0.0000 |
| Rd_Configuration | 4.0000 | 17.1126 | 4.2781 | 0.2371 | 0.9175 |
| Rd_Surface | 4.0000 | 188.2856 | 47.0714 | 2.6091 | 0.0337 |
| Rd_Conditions | 3.0000 | 20.4264 | 6.8088 | 0.3774 | 0.7693 |
| Light | 5.0000 | 661.0898 | 132.2180 | 7.3286 | 0.0000 |
| Weather | 4.0000 | 119.5621 | 29.8905 | 1.6568 | 0.1570 |
| Traffic_Control | 4.0000 | 400.3357 | 100.0839 | 5.5475 | 0.0002 |
| Work_Area | 1.0000 | 105.6916 | 105.6916 | 5.8583 | 0.0155 |
| Residual | 23078.0000 | 416355.9220 | 18.0412 | NaN | NaN |

Tabla 21: ANOVA para modelo 0

## Modelo 1

| Variable | df | sum_sq | mean_sq | F | $P (¿F$) |
|---|---|---|---|---|---|
| year | 5.0000 | 2.6001 | 0.5200 | 0.4781 | 0.7929 |
| Month | 11.0000 | 4.5389 | 0.4126 | 0.3794 | 0.9646 |
| Time_of_Day | 5.0000 | 69.2948 | 13.8590 | 12.7423 | 0.0000 |
| Rd_Feature | 4.0000 | 165.8339 | 41.4585 | 38.1180 | 0.0000 |
| Rd_Character | 6.0000 | 30.6552 | 5.1092 | 4.6975 | 0.0001 |
| Rd_Class | 2.0000 | 111.0778 | 55.5389 | 51.0639 | 0.0000 |
| Rd_Configuration | 4.0000 | 0.9758 | 0.2439 | 0.2243 | 0.9250 |
| Rd_Surface | 4.0000 | 15.9737 | 3.9934 | 3.6717 | 0.0054 |
| Rd_Conditions | 3.0000 | 1.4278 | 0.4759 | 0.4376 | 0.7261 |
| Light | 5.0000 | 55.8626 | 11.1725 | 10.2723 | 0.0000 |
| Weather | 4.0000 | 8.6196 | 2.1549 | 1.9813 | 0.0944 |
| Traffic_Control | 4.0000 | 38.0889 | 9.5222 | 8.7550 | 0.0000 |
| Work_Area | 1.0000 | 2.0712 | 2.0712 | 1.9043 | 0.1676 |
| Residual | 23078.0000 | 25100.4577 | 1.0876 | NaN | NaN |

Tabla 23: ANOVA para modelo 1

## Modelo 2

| Variable | df | sum_sq | mean_sq | F | $P (¿F$) |
|---|---|---|---|---|---|
| Time_of_Day | 5.0000 | 68.4482 | 13.6896 | 12.5886 | 0.0000 |
| Rd_Feature | 4.0000 | 166.2881 | 41.5720 | 38.2283 | 0.0000 |
| Rd_Character | 6.0000 | 30.1610 | 5.0268 | 4.6225 | 0.0001 |
| Rd_Class | 2.0000 | 111.2879 | 55.6439 | 51.1684 | 0.0000 |
| Rd_Surface | 4.0000 | 15.0100 | 3.7525 | 3.4507 | 0.0080 |
| Light | 5.0000 | 51.9180 | 10.3836 | 9.5484 | 0.0000 |
| Traffic_Control | 4.0000 | 37.3621 | 9.3405 | 8.5893 | 0.0000 |
| Residual | 23106.0000 | 25127.0029 | 1.0875 | NaN | NaN |

Tabla 25: ANOVA para modelo 2

## Modelo 3

| Variable | df | sum_sq | mean_sq | F | $P (¿F$) |
|---|---|---|---|---|---|
| Rd_Class | 2.0000 | 174.9344 | 87.4672 | 80.6077 | 0.0000 |
| Traffic_Control | 4.0000 | 112.3334 | 28.0833 | 25.8809 | 0.0000 |
| Time_of_Day | 5.0000 | 65.1064 | 13.0213 | 12.0001 | 0.0000 |
| Light | 5.0000 | 51.4740 | 10.2948 | 9.4874 | 0.0000 |
| Rd_Feature | 4.0000 | 39.8064 | 9.9516 | 9.1712 | 0.0000 |
| Rd_Character | 6.0000 | 23.6664 | 3.9444 | 3.6351 | 0.0013 |
| Rd_Surface | 4.0000 | 13.1543 | 3.2886 | 3.0307 | 0.0165 |
| : Time_of_Day : Light | 25.0000 | 41.8994 | 1.6760 | 1.5445 | 0.0404 |
| : Rd_Feature : Rd_Character | 24.0000 | 39.1408 | 1.6309 | 1.5030 | 0.0542 |
| : Rd_Feature : Rd_Surface | 16.0000 | 30.5494 | 1.9093 | 1.7596 | 0.0304 |
| : Rd_Character : Rd_Surface | 24.0000 | 18.2593 | 0.7608 | 0.7011 | 0.8559 |
| : Rd_Feature : Rd_Character : Rd_Surface | 96.0000 | 120.3806 | 1.2540 | 1.1556 | 0.1419 |
| Residual | 22979.0000 | 24934.4466 | 1.0851 | NaN | NaN |

Tabla 27: ANOVA para modelo 3

## Modelo 4

| Variable | df | sum_sq | mean_sq | F | $P (¿F$) |
|---|---|---|---|---|---|
| Rd_Class | 2.0000 | 174.9344 | 87.4672 | 80.7151 | 0.0000 |
| Traffic_Control | 4.0000 | 112.3334 | 28.0833 | 25.9154 | 0.0000 |
| Rd_Feature | 4.0000 | 40.4497 | 10.1124 | 9.3318 | 0.0000 |
| Rd_Character | 6.0000 | 29.5477 | 4.9246 | 4.5445 | 0.0001 |
| Rd_Surface | 4.0000 | 13.3979 | 3.3495 | 3.0909 | 0.0149 |
| Time_of_Day | 5.0000 | 61.2545 | 12.2509 | 11.3052 | 0.0000 |
| Light | 5.0000 | 48.5578 | 9.7116 | 8.9619 | 0.0000 |
| : Rd_Feature : Rd_Character | 24.0000 | 39.9091 | 1.6629 | 1.5345 | 0.0456 |
| : Rd_Feature : Rd_Surface | 16.0000 | 26.0955 | 1.6310 | 1.5051 | 0.0879 |
| : Rd_Character : Rd_Surface | 24.0000 | 22.6416 | 0.9434 | 0.8706 | 0.6450 |
| : Rd_Feature : Time_of_Day | 20.0000 | 27.7317 | 1.3866 | 1.2795 | 0.1799 |
| : Rd_Character : Time_of_Day | 30.0000 | 38.7420 | 1.2914 | 1.1917 | 0.2165 |
| : Rd_Surface : Time_of_Day | 20.0000 | 30.8236 | 1.5412 | 1.4222 | 0.0994 |
| : Rd_Feature : Light | 20.0000 | 30.3820 | 1.5191 | 1.4018 | 0.1087 |
| : Rd_Character : Light | 30.0000 | 50.4859 | 1.6829 | 1.5530 | 0.0274 |
| : Rd_Surface : Light | 20.0000 | 18.9922 | 0.9496 | 0.8763 | 0.6186 |
| : Rd_Feature : Rd_Character : Rd_Surface | 96.0000 | 106.9679 | 1.1142 | 1.0282 | 0.4047 |
| : Rd_Feature : Rd_Character : Time_of_Day | 120.0000 | 127.7655 | 1.0647 | 0.9825 | 0.5371 |
| : Rd_Feature : Rd_Surface : Time_of_Day | 80.0000 | 111.8225 | 1.3978 | 1.2899 | 0.0419 |
| : Rd_Character : Rd_Surface : Time_of_Day | 120.0000 | 120.3298 | 1.0027 | 0.9253 | 0.7085 |
| : Rd_Feature : Rd_Character : Light | 120.0000 | 135.3034 | 1.1275 | 1.0405 | 0.3627 |
| : Rd_Feature : Rd_Surface : Light | 80.0000 | 82.2389 | 1.0280 | 0.9486 | 0.6093 |
| : Rd_Character : Rd_Surface : Light | 120.0000 | 124.2919 | 1.0358 | 0.9558 | 0.6193 |
| : Rd_Feature : Rd_Character : Rd_Surface : Tim... | 480.0000 | 484.3093 | 1.0090 | 0.9311 | 0.8557 |
| : Rd_Feature : Rd_Character : Rd_Surface : Light | 480.0000 | 594.1913 | 1.2379 | 1.1423 | 0.0177 |
| Residual | 22481.0000 | 24361.5993 | 1.0837 | NaN | NaN |

Tabla 29: ANOVA para modelo 4

### Modelo 5

| Variable | df | sum_sq | mean_sq | F | $P (¿F$) |
|---|---|---|---|---|---|
| Rd_Class | 2.0000 | 174.9344 | 87.4672 | 80.7193 | 0.0000 |
| Traffic_Control | 4.0000 | 112.3334 | 28.0833 | 25.9168 | 0.0000 |

Continued on next page

| Variable | df | sum_sq | mean_sq | F | $P (¿F$) |
|---|---|---|---|---|---|
| Rd_Feature | 4.0000 | 40.4497 | 10.1124 | 9.3323 | 0.0000 |
| Rd_Character | 6.0000 | 29.5477 | 4.9246 | 4.5447 | 0.0001 |
| Rd_Surface | 4.0000 | 13.3979 | 3.3495 | 3.0911 | 0.0149 |
| Time_of_Day | 5.0000 | 61.2545 | 12.2509 | 11.3058 | 0.0000 |
| Light | 5.0000 | 48.5578 | 9.7116 | 8.9623 | 0.0000 |
| : Rd_Feature : Rd_Character | 24.0000 | 39.9091 | 1.6629 | 1.5346 | 0.0456 |
| : Rd_Feature : Rd_Surface | 16.0000 | 26.0955 | 1.6310 | 1.5051 | 0.0878 |
| : Rd_Feature : Time_of_Day | 20.0000 | 27.0867 | 1.3543 | 1.2499 | 0.2017 |
| : Rd_Character : Time_of_Day | 30.0000 | 27.9352 | 0.9312 | 0.8593 | 0.6862 |
| : Rd_Surface : Time_of_Day | 20.0000 | 33.3872 | 1.6694 | 1.5406 | 0.0578 |
| : Rd_Feature : Light | 20.0000 | 29.9218 | 1.4961 | 1.3807 | 0.1190 |
| : Rd_Character : Light | 30.0000 | 47.6300 | 1.5877 | 1.4652 | 0.0483 |
| : Rd_Feature : Rd_Character : Rd_Surface | 120.0000 | 131.8752 | 1.0990 | 1.0142 | 0.4397 |
| : Rd_Feature : Rd_Surface : Time_of_Day | 80.0000 | 112.4661 | 1.4058 | 1.2974 | 0.0385 |
| : Rd_Feature : Rd_Character : Light | 120.0000 | 134.2030 | 1.1184 | 1.0321 | 0.3867 |
| : Rd_Feature : Rd_Character : Rd_Surface : Light | 700.0000 | 681.6559 | 0.9738 | 0.8987 | 0.9725 |
| Residual | 22666.0000 | 24560.8083 | 1.0836 | NaN | NaN |

Tabla 31: ANOVA para modelo 5

### Modelo 6

| Variable | df | sum_sq | mean_sq | F | $P (¿F$) |
|---|---|---|---|---|---|
| Rd_Class | 2.0000 | 174.9344 | 87.4672 | 80.7317 | 0.0000 |
| Traffic_Control | 4.0000 | 112.3334 | 28.0833 | 25.9208 | 0.0000 |
| Rd_Feature | 4.0000 | 40.4497 | 10.1124 | 9.3337 | 0.0000 |
| Rd_Character | 6.0000 | 29.5477 | 4.9246 | 4.5454 | 0.0001 |
| Rd_Surface | 4.0000 | 13.3979 | 3.3495 | 3.0916 | 0.0148 |
| Time_of_Day | 5.0000 | 61.2545 | 12.2509 | 11.3075 | 0.0000 |
| Light | 5.0000 | 48.5578 | 9.7116 | 8.9637 | 0.0000 |
| : Rd_Feature : Rd_Character | 24.0000 | 39.9091 | 1.6629 | 1.5348 | 0.0456 |
| : Rd_Feature : Rd_Surface | 16.0000 | 26.0955 | 1.6310 | 1.5054 | 0.0878 |
| : Rd_Surface : Time_of_Day | 20.0000 | 32.7552 | 1.6378 | 1.5116 | 0.0663 |
| : Rd_Character : Light | 30.0000 | 44.8885 | 1.4963 | 1.3811 | 0.0802 |
| : Rd_Feature : Rd_Surface : Time_of_Day | 100.0000 | 139.8376 | 1.3984 | 1.2907 | 0.0271 |

Continued on next page

| Variable | df | sum_sq | mean_sq | F | $P (¿F$) |
|---|---|---|---|---|---|
| : Rd_Feature : Rd_Character : Rd_Surface : Light | 960.0000 | 1056.2630 | 1.1003 | 1.0155 | 0.3646 |
| Residual | 22693.0000 | 24586.2776 | 1.0834 | NaN | NaN |

Tabla 33: ANOVA para modelo 6

## Anexo 2: código en Python de los problemas

```
1
2  # ## Librerias y definiciones necesarias
3  import numpy as np
4  import matplotlib.pyplot as plt
5  import pandas as pd
6  import statsmodels.api as sm
7  import statsmodels.formula.api  as smf
8  import os
9  if not(os.path.isdir("tarea")):
10     os.mkdir("tarea")
11
12 def cleanVarName(var):
13     if len(var) > 1:
14         if var[0]=="C" and var[1]=="(":
15             if len(var.split("[")) > 1:
16                 return var.split(",")[0][2:] + " [" + var.split
    ("[")[1]
17             else:
18                 return var.split(",")[0][2:]
19         elif len(var.split(":")) > 1:
20             fin = ""
21             for v in var.split(":"):
22                 fin += " : " + v
23             return fin
24         else:
25             return var
26     else:
27         return var
28 def significanceTable(fit):
29     df = pd.DataFrame(columns = ["R-squared",
30                                  "AIC",
31                                  "BIC",
32                                  "Log-Likelihood",
33                                 "F-statistic",
```

```
                                    "Prob (F-statistic)"])
    df.loc[0]=[fit.rsquared,
    fit.aic,
    fit.bic,
    fit.llf,
    fit.fvalue,
    fit.f_pvalue]
    return df
def coefficientTable(fit):
    df = fit.summary2().tables[1]
    df = df.rename(columns={"P>|t|":"$P (> |t|)$"})

    df.index = [cleanVarName(var) for var in df.index]
    return df
def anovaTable(fit):
    df = sm.stats.anova_lm(fit)
    df = df.rename(columns={"PR(>F)":"$P (> F$)"})
    df.index = [cleanVarName(var) for var in df.index]
    df.index.name = "Variable"
    return df
def comparisonTable(fits):
    dfs = [significanceTable(fit) for fit in fits]
    df = dfs[0]
    for i in range(1,len(dfs)):
        df.loc[i] = dfs[i].iloc[0]
    formulas = [fit.model.formula for fit in fits]
    df.insert(loc=0,column="Model",value=formulas)
    return df


# ## Ejercicio 1. Problema 10
df = pd.read_csv("carseats.csv",header=0)
df.head().to_latex("tarea/1-data.tex")

fit = smf.ols(formula="Sales ~ Price + C(Urban) + C(US)",
              data=df).fit()
print(fit.summary())

coefficientTable(fit)

coefficientTable(fit).to_latex(buf="tarea/1-mod1.tex")

fit2 = smf.ols(
    formula="Sales ~ Price + C(US)",
    data=df).fit()
print(fit2.summary())
```

```
80
81 coefficientTable(fit2).to_latex(buf="tarea/1-mod2.tex")
82 comparisonTable([fit,fit2]).to_latex(buf="tarea/1-modCompar.tex
   ")
83
84 abs(fit.rsquared - fit2.rsquared)
85
86 print(fit2.conf_int(alpha = 0.05))
87
88 a=0.5
89 fig = plt.figure(figsize=(6,4))
90 plt.scatter(fit2.predict(),fit2.resid.tolist(),alpha = a)
91 plt.ylabel("Residuals")
92 plt.xlabel("Fitted Sales")
93 plt.title("Residual Plot")
94 plt.grid()
95 plt.tight_layout()
96 plt.savefig("tarea/1-resplot.pdf")
97 plt.show()
98
99 fig = plt.figure(figsize=(10,4))
100 n = len(fit2.resid.tolist())
101 plt.scatter(range(n),fit2.resid.tolist(), alpha = a)
102 plt.plot([0,n],[0,0],c="k",lw=3,ls="--", alpha = a)
103 plt.ylabel("Residuals")
104 plt.xlabel("Index")
105 plt.title("Residual Series")
106 plt.grid()
107 plt.tight_layout()
108 plt.savefig("tarea/1-restimeplot.pdf")
109 plt.show()
110
111 fig = plt.figure(figsize=(12,8))
112 fig = sm.graphics.plot_ccpr_grid(fit2,fig=fig)
113 for ax in fig.axes:
114     ax.grid()
115 fig.suptitle("Component-Component Residuals",x=0.5,y=1.05,
    fontsize=20,ha="center")
116 plt.tight_layout()
117 fig.savefig("tarea/1-compcomp.pdf",bbox_inches="tight")
118
119 fig = plt.figure(figsize=(12,8))
120 fig = sm.graphics.plot_regress_exog(fit2,"Price",fig=fig)
121 for ax in fig.axes:
122     ax.grid()
123 plt.tight_layout()
```

```
124 plt.savefig("tarea/1-regdiag.pdf")
125
126 fig, ax = plt.subplots(figsize=(12,8))
127 fig = sm.graphics.influence_plot(fit2, ax=ax)
128 plt.grid()
129 plt.tight_layout()
130 plt.savefig("tarea/1-influence.pdf")
131 plt.show()
132
133 sm.graphics.qqplot(fit2.resid)
134 plt.title("Q-Q plot of predicted values")
135 plt.grid()
136 plt.tight_layout()
137 plt.savefig("tarea/1-qqplot.pdf")
138 plt.show()
139
140 df.iloc[[42,174,165],:].to_latex(buf="tarea/1-outliers.tex")
141
142
143 # ## Ejercicio 1. Problema 14
144 np.random.seed(1)
145 x1 = np.random.uniform(size=100)
146 x2 = 0.5*x1 + np.random.randn(100)/10
147 y = 2 + 2*x1 +0.3*x2 + np.random.randn(100)
148 df = pd.DataFrame()
149 df["x1"] = x1
150 df["x2"] = x2
151 df["y"] = y
152 df.head().to_latex(buf="tarea/2-data.tex")
153 print(df.head())
154
155 df.plot.scatter(x="x1",y="x2")
156 plt.grid()
157 plt.title(u"Correlaci n entre x1 y x2")
158 plt.savefig("tarea/2-corr.pdf")
159 cor = np.corrcoef(x1,x2)[0][1]
160 plt.text(0.8,0.0,"R^2={0}".format(round(cor,5)),
161          va = "top",
162          ha = "center",
163     bbox=dict(boxstyle='round', facecolor='wheat', alpha=0.5))
164 plt.tight_layout()
165 plt.savefig("tarea/2-corr.pdf")
166 plt.show()
167
168 formulas = ["y ~ x1 +x2", "y ~ x1", "y ~x2"]
169 fits = [ smf.ols(formula = form , data=df).fit() for form in
```

```
   formulas]
for fit in fits:
    print(fit.summary())
    print()
    print()
    print(fit.pvalues)
    print()
    print()
    coefficientTable(fit).to_latex(buf = "tarea/2-mod{0}.tex".
   format(fits.index(fit)+1))

comparisonTable(fits).to_latex(buf="tarea/2-modComp.tex",index=
   False)

df.loc[100] = [0.1,0.8,0.6]
df.tail()

df.iloc[1:100].plot.scatter(x="x1",y="x2",label="data")
plt.scatter([0.1],[0.8],c="C1",label="New point")
plt.legend()
plt.grid()
plt.title("Plano x-y")
plt.tight_layout()
plt.savefig("tarea/2-newData.pdf")
plt.show()

fit.resid

fits = [ smf.ols(formula = form , data=df).fit() for form in
   formulas]
for fit in fits:
    print(fit.summary())
    print()
    print()
    print(fit.pvalues)
    print()
    print()
    print( fit.resid[100] / ( fit.resid.std() / np.sqrt(101) )
   )
    coefficientTable(fit).to_latex(buf = "tarea/2-newMod{0}.tex
   ".format(fits.index(fit)+1))


# ## Problema 2
df = pd.read_csv("June_13_data.csv",header=0)
df=df.astype({'year': 'object',"Month":"object","Time_of_Day":"
```

```
    object"})
df

par = ["p{2cm}" for i in range(10)]
colFormat = "ccccc"
for p in par:
    colFormat += p
df.head().to_latex(buf="tarea/3-data.tex",column_format=
    colFormat)

cols = df.columns.tolist()
for col in cols:
    if col != "Crash_Score":
        nb=30
        fig, ax = plt.subplots(ncols=2,nrows=1,figsize=(12,6))
        df.boxplot(column="Crash_Score",
                           by=col,
                           fontsize=16,
                           figsize=(10,10),
                           rot = 45,
                           ax = ax[0]
        )
        for value in set(df[col]):
            df[df[col]==value]["Crash_Score"].hist(alpha=0.5,
                                                   ax=ax[1],
                                                   label=str(
    value),
                                                   bins = nb,
                                                   density=True
            )
        ax[1].legend()
        ax[1].set_title("Histogram")
        plt.tight_layout()
        plt.savefig("tarea/3-{0}Hist.pdf".format(col))
        plt.show()
        plt.close()
        """
        hist = df.hist(column="Crash_Score",
                       by=col,
                       figsize=(8,8),
                       bins=nb,
                       sharex=True,
                       sharey=True,
                       density = True,
        )
        for a in hist.flat:
```

```
            a.grid()
        plt.tight_layout()
        plt.show()
        plt.close()
        """

df1 = df[["Crash_Score","year","Month","Time_of_Day"]]
df1["time"]=(df1["year"]-df1["year"].min())*12*6 + (df1["Month"]-1)*6 + df["Time_of_Day"]-1
df1.plot.scatter(y="Crash_Score",x="time",figsize=(14,6),alpha=0.3)
times = list(set(df1["time"]))
cs_avg = []
for t in times:
    avg = df1[df1["time"]==t]["Crash_Score"].mean()
    cs_avg.append(avg)
plt.plot(times,cs_avg,color = "C1",alpha=0.8)
plt.grid()
plt.title("Serie de tiempo",fontsize=24)
plt.xlabel("Tiempo",fontsize=20)
plt.ylabel("Crash_Score",fontsize=20)
plt.xticks(fontsize=18)
plt.yticks(fontsize=18)
plt.tight_layout()
plt.savefig("tarea/3-timeplot.pdf")
plt.show()

refers = {
    "year":"2014",
    "Month":"10",
    "Time_of_Day":"4",
    "Rd_Feature":"NONE",
    "Rd_Character":"STRAIGHT-LEVEL",
    "Rd_Class":"OTHER",
    "Rd_Configuration":"TWO-WAY-UNPROTECTED-MEDIAN",
    "Rd_Surface":"SMOOTH ASPHALT",
    "Rd_Conditions":"OTHER",
    "Light":"DAYLIGHT",
    "Weather":"CLEAR",
    "Traffic_Control":"NONE",
    "Work_Area":"NO"
}

cols = df.columns.tolist()
model = "Crash_Score ~ "
for col in cols:
```

```
297     if col != "Crash_Score":
298         if col == "year" or col =="Month" or col=="Time_of_Day"
    :
299             model+=(" + C({0},Treatment({1}))".format(col,
    refers[col]))
300         else:
301             model+=(" + C({0},Treatment('{1}'))".format(col,
    refers[col]))
302
303
304 print("Model to try:")
305 print()
306 print(model)
307
308 models = []
309 mod = smf.ols(formula=model,data=df)
310 models.append(mod.fit())
311
312 print(models[-1].summary().tables[0])
313 print(anovaTable(models[-1]))
314
315 import scipy.stats as st
316
317 st.boxcox_normplot(df["Crash_Score"],-3.0,4.0,plot=plt)
318 plt.grid()
319 plt.tight_layout()
320 plt.savefig("tarea/3-boxcox.pdf")
321 plt.show()
322
323 st.boxcox_normmax(df["Crash_Score"],brack=(-3.0,4.0))
324
325 lmax = st.boxcox(df["Crash_Score"])[1]
326 df1 = df.copy()
327 df1["Crash_Score"] = st.boxcox(df1["Crash_Score"],lmax)
328 df1.rename(columns={'Crash_Score':'Crash_Score_boxcox_0_27'},
    inplace=True)
329
330 cols = df1.columns.tolist()
331 model = "Crash_Score_boxcox_0_27 ~ "
332 for col in cols:
333     if col != "Crash_Score_boxcox_0_27":
334         if col == "year" or col =="Month" or col=="Time_of_Day"
    :
335             model+=(" + C({0},Treatment({1}))".format(col,
    refers[col]))
336         else:
```

```
337             model+=(" + C({0},Treatment('{1}'))".format(col,
    refers[col]))
338
339 print("Model to try:")
340 print()
341 print(model)
342
343 mod = smf.ols(formula=model,data=df1)
344 models.append(mod.fit())
345 print(models[-1].summary().tables[0])
346 print(anovaTable(models[-1]))
347
348 necesary = ["Time_of_Day",
349 "Rd_Feature",
350 "Rd_Character",
351 "Rd_Class",
352 "Rd_Surface",
353 "Light",
354 "Traffic_Control"]
355 model = "Crash_Score_boxcox_0_27 ~ "
356 model1= ""
357 for col in necesary:
358     if col == "year" or col =="Month" or col=="Time_of_Day":
359         model+=(" + C({0},Treatment({1}))".format(col,refers[
    col]))
360     else:
361         model+=(" + C({0},Treatment('{1}'))".format(col,refers[
    col]))
362 model1 = model1[1:]
363
364 print("Model to try:")
365 print()
366 print(model)
367
368 mod = smf.ols(formula=model,data=df1)
369 models.append(mod.fit())
370 models[-1].summary().tables[0]
371
372 necesary = [#"Time_of_Day",
373 #"Rd_Feature",
374 #"Rd_Character",
375 "Rd_Class",
376 #"Rd_Surface",
377 #"Light",
378 "Traffic_Control"]
379 inter1 = "Time_of_Day*Light"
```

```
inter2 = "Rd_Feature*Rd_Character*Rd_Surface"
model = "Crash_Score_boxcox_0_27 ~ "
model1= ""
for col in necesary:
    if col == "year" or col =="Month" or col=="Time_of_Day":
        model+=(" + C({0},Treatment({1}))".format(col,refers[
   col]))
    else:
        model+=(" + C({0},Treatment('{1}'))".format(col,refers[
   col]))

model1 = inter1 + " + " + inter2
model += " + " + model1

print("Model to try:")
print()
print(model)

mod = smf.ols(formula=model,data=df1)
models.append(mod.fit())

models[-1].summary().tables[0]

sm.stats.anova_lm(models[-1])

necesary = [#"Time_of_Day",
#"Rd_Feature",
#"Rd_Character",
"Rd_Class",
#"Rd_Surface",
#"Light",
"Traffic_Control"]
inter1 = "Rd_Feature*Rd_Character*Rd_Surface*Time_of_Day"
inter2 = "Rd_Feature*Rd_Character*Rd_Surface*Light"
model = "Crash_Score_boxcox_0_27 ~ "
model1= ""
for col in necesary:
    if col == "year" or col =="Month" or col=="Time_of_Day":
        model+=(" + C({0},Treatment({1}))".format(col,refers[
   col]))
    else:
        model+=(" + C({0},Treatment('{1}'))".format(col,refers[
   col]))

model1 = inter1 + " + " + inter2
model += " + " + model1
```

```
422 print("Model to try:")
423 print()
424 print(model)
425 
426 mod = smf.ols(formula=model,data=df1)
427 models.append(mod.fit())
428 
429 models[-1].summary().tables[0]
430 
431 tab = sm.stats.anova_lm(models[-1])
432 print(tab)
433 
434 sm.graphics.qqplot(models[-1].resid)
435 plt.title("Q-Q plot of predicted values")
436 plt.grid()
437 plt.tight_layout()
438 plt.savefig("tarea/3-qqplot4.pdf")
439 plt.show()
440 
441 model = ""
442 for var in tab.index.tolist():
443     if tab.loc[var,"PR(>F)"]<= 0.5:
444         model += "+ " + var
445 model = "Crash_Score_boxcox_0_27 ~ " + model
446 
447 print("Model to try:")
448 print()
449 print(model)
450 
451 mod = smf.ols(formula=model,data=df1)
452 models.append(mod.fit())
453 
454 models[-1].summary().tables[0]
455 
456 model = ""
457 for var in tab.index.tolist():
458     if tab.loc[var,"PR(>F)"]<= 0.1:
459         model += "+ " + var
460 model = "Crash_Score_boxcox_0_27 ~ " + model
461 
462 print("Model to try:")
463 print()
464 print(model)
465 
466 mod = smf.ols(formula=model,data=df1)
467 models.append(mod.fit())
```

```
468
469 models[-1].summary().tables[0]
470
471 for model in models:
472     i = models.index(model)
473     coefficientTable(model).to_latex(buf="tarea/3-mod{0}Cof.tex
    ".format(i),column_format="p{4cm}cccccc",longtable=True,
    float_format = "%.4f")
474     anovaTable(model).to_latex(buf="tarea/3-mod{0}Anova.tex".
    format(i),column_format="p{6cm}lccccc",longtable=True,
    float_format = "%.4f")
475
476 tab = comparisonTable(models).drop("Model",axis="columns")
477 tab.index.name="model"
478 tab.to_latex(buf="tarea/3-comparison.tex",column_format="
    lrrrrrr")
```

## Referencias

[1] Hastie et al. *An Introduction to Statitistical Learning.* Editorial Springer. Séptima edición. 2013.