Estudio y agrupación de clases de homicidios en Colombia en los últimos 8 años.

Nombre de la organización: Policía Nacional de Colombia.

Autores: Elkin Rafael Pulido Farias – Jorge Duban Bernal Cardenas.

Información general sobre el conjunto de datos

Tablero de control de los datos utilizados

	Cantidad de ejemplos o instancias utilizadas
Entrenamiento	71.613
Validación	15.346
Prueba	15.346
Total	102.305

Modelos evaluados

Modelos evaluados

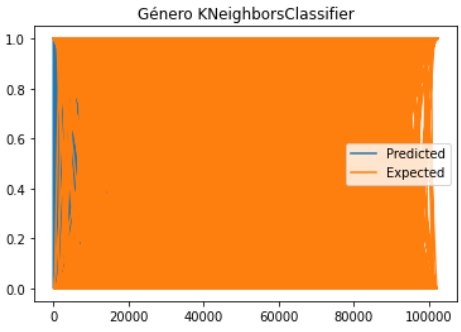
Objetivo		**Criterio de evaluación**		
Predecir el genero al que se le cometió el homicidio		Obtener un modelo con mínimo el 90% de precisión, tanto en entrenamiento como en evaluación.		
Métrica		Accuracy		
Nombre	**Algoritmo(s) principal(es)**	**K-fold**	**Tasa entrenamiento**	**Tasa evaluación**
modelo_generoLoR.pickle	LogisticRegression	K=10	0.915	0.918
modelo_generoKnn.pickle	KNeighborsClassifier	K=10	0.918	0.910

```
y_predictloR = logReg.predict(x_val)
#grafica para el modelo logReg
import matplotlib.pyplot as plt
plt.plot(y_predictloR, label='Predicted')
plt.plot(y_test.values, label='Expected')
plt.title("Género LogisticRegression")
plt.legend()
plt.show()
```

✓ 0 s se ejecutó 20:56

Gráfica 1 comparación predicción-valor esperado para LogisticRegression de Genero

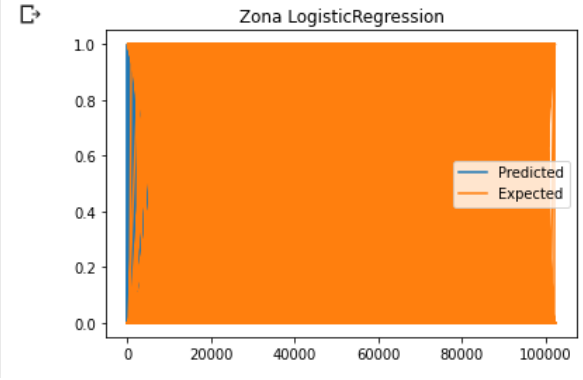
```
y_predicknc = knn.predict(x_val)
#grafica para el modelo logReg
import matplotlib.pyplot as plt
plt.plot(y_predicknc, label='Predicted')
plt.plot(y_test, label='Expected')
plt.title("Género KNeighborsClassifier")
plt.legend()
plt.show()
```



Gráfica 2 comparación predicción-valor esperado para KNeighborsClassifier de Genero

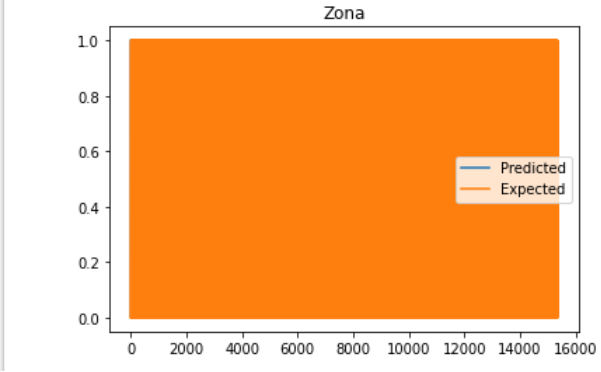
Objetivo		Criterio de evaluación		
Predecir la zona en la que se cometió un homicidio		Obtener un modelo con mínimo el 90% de precisión, tanto en entrenamiento como en evaluación.		
Métrica		Accuracy		
Nombre	**Algoritmo(s) principal(es)**	**K-fold**	**Tasa entrenamiento**	**Tasa evaluación**
modelo_zonaLoR.pickle	LogisticRegression	K=10	0.759	0.758
modelo_zonaKnn.pickle	KNeighborsClassifier	K=10	0.8140	0.7360
modelo_zonaSvc.pickle	SVC	No se usa	0.773	0.755

```
y_predictRl = logReg.predict(x_val)
import matplotlib.pyplot as plt
plt.plot(y_predictRl, label='Predicted')
plt.plot(y_test, label='Expected')
plt.title("Zona LogisticRegression")
plt.legend()
plt.show()
```



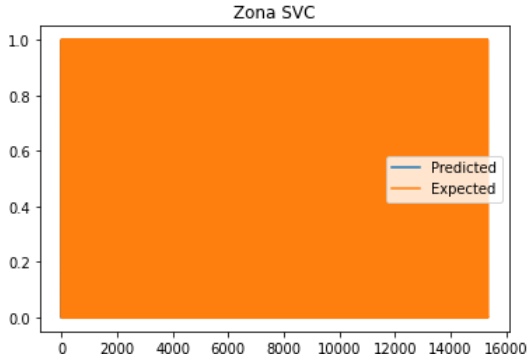
Gráfica 3 comparación predicción-valor esperado para LogisticRegression de Zona

```
y_predictknn = knn.predict(x_val)
import matplotlib.pyplot as plt
plt.plot(y_predictknn, label='Predicted')
plt.plot(y_test.values, label='Expected')
plt.title("Zona")
plt.legend()
plt.show()
```



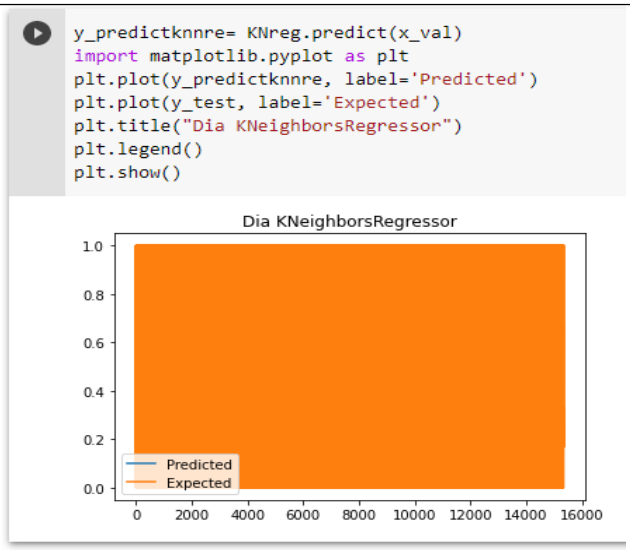
Gráfica 4 comparación predicción-valor esperado para KNeighborsClassifier de Zona

```
y_predictsvc = svm.predict(x_val)
import matplotlib.pyplot as plt
plt.plot(y_predictsvc, label='Predicted')
plt.plot(y_test.values, label='Expected')
plt.title("Zona SVC")
plt.legend()
plt.show()
```



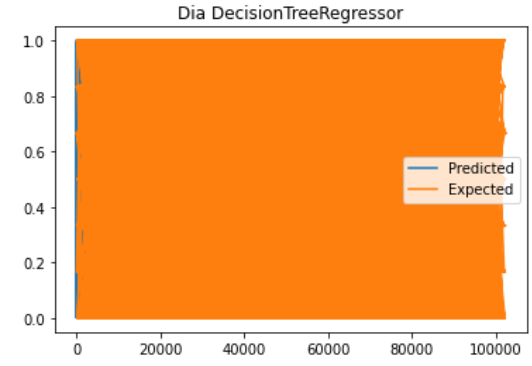
Gráfica 5 comparación predicción-valor esperado para SVC de Zona

Objetivo		Criterio de evaluación		
Predecir el día en que se produjo un homicidio.		Obtener un modelo con mínimo el 90% de precisión, tanto en entrenamiento como en evaluación.		
Métrica		Accuracy		
Nombre	**Algoritmo(s) principal(es)**	**K-fold**	**Tasa entrenamiento**	**Tasa evaluación**
modelo_diaKNreg.pickle	KNeighborsRegressor	No se usa	0.25	0.13
modelo_diaTreeR.pickle	DecisionTreeRegressor	No se usa	0.915	0.910



Gráfica 6 comparación predicción-valor esperado para KNeighborsRegressor de Día

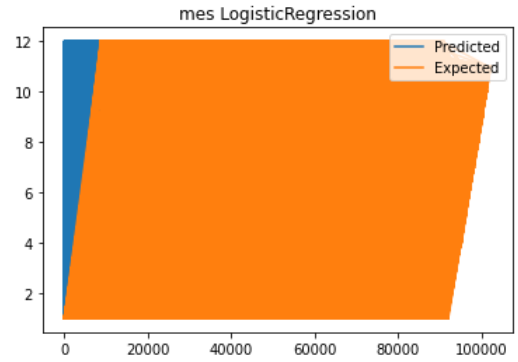
```
y_predictreeR= treeR.predict(x_val)
import matplotlib.pyplot as plt
plt.plot(y_predictreeR, label='Predicted')
plt.plot(y_test, label='Expected')
plt.title("Dia DecisionTreeRegressor")
plt.legend()
plt.show()
```



Gráfica7 comparación predicción-valor esperado para DecisionTreeRegressor de Día

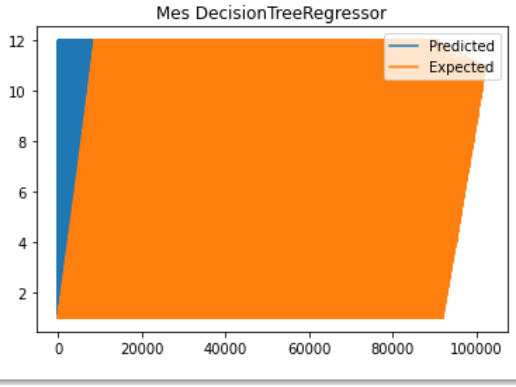
Objetivo		**Criterio de evaluación**		
Predecir el mes en que se cometió un homicidio.		Obtener un modelo con mínimo el 90% de precisión, tanto en entrenamiento como en evaluación.		
Métrica		Accuracy		
Nombre	**Algoritmo(s) principal(es)**	**K-fold**	**Tasa entrenamiento**	**Tasa evaluación**
modelo_mesLoRe.pickle	LogisticRegression		0.118	0.109
modelo_mesTreeR.pickle	DecisionTreeRegressor		0.86	0.82

```
y_prediclore = logReg.predict(x_val)
#grafica para el modelo logReg
import matplotlib.pyplot as plt
plt.plot(y_prediclore, label='Predicted')
plt.plot(y_test, label='Expected')
plt.title("mes LogisticRegression")
plt.legend()
plt.show()
```



Gráfica 8 comparación predicción-valor esperado para LogisticRegression del mes

```
plt.plot(y_predicTreeRt, label='Predicted')
plt.plot(y_test, label='Expected')
plt.title("Mes DecisionTreeRegressor")
plt.legend()
plt.show()
```



Gráfica 9 comparación predicción-valor esperado para DecisionTreeRegressor del mes

Lecciones aprendidas

- **¿Existen formas de simplificar o mejorar alguna de las fases u operación particular?, ¿cuáles?**

  Notamos que mientras comenzábamos a prepararnos para hacer los respectivos modelos, algunos de los objetivos planteados en la primera fase (compresión del negocio) no eran muy específicos o no se podían utilizar para hacer un respectivo modelo idóneo y por lo tanto, teniendo lo anterior presente, no se propusieron, en su mayoría, objetivos adecuados para una predicción que estuvieran a la par con el objetivo general del proyecto de lograr predecir los homicidios para el año 2019 con una presión de mínimo el 90%.

  Por otro lado, se llegó a la conclusión de se debe hacer un mejor estudio y revisión de los datos para determinar si algunos de estos están sesgados en el entendimiento de os datos.

- **¿Cuáles fueron los fallos o errores de cometieron?, ¿Cómo se pueden evitar la próxima vez?**

  Durante el proceso de entrenamiento notamos que en algunos modelos la maquina no podía cumplir con el proceso de ejecución del algoritmo SVC o de la misma manera se tardaba mucho a pesar de que la cantidad de datos no es muy grande para su proceso, por tal razón se requerirá en un próximo proyecto tener presente desde la fase uno el equipo necesario y competente así como los recursos necesarios (como una buena y estable conexión a internet) para cumplir adecuadamente con la tarea para no tener retrasos en el cronograma o estar presionado por la fecha de finalización de la respectiva fase de modelado.

- **¿Hay callejones sin salida, como modelos específicos que no ofrecen ningún resultado?, ¿se pudo evitar?, ¿cómo?**

  En algunos modelos, para las predicciones de los objetivos que se agregaron como la predicción del mes y día en que se produjo un homicidio, no cumplieron con el criterio de evaluación ya que su porcentaje de éxito fue muy bajo, menos al 50%; lo más probable es que es debido a que en estos modelos la cantidad de atributos o clasificaciones es mayor y en el data set están sesgados, por lo tanto, no se encontró una manera rápida y eficiente que permitiera mejorar ese porcentaje, además, de recurrir a la primera fase y hacer una mejor investigación de los datos para ver exactamente si están sesgados.

  Por otro lado, en la graficación de los resultados de los modelos, sus resultados visuales fueron iguales en algunos algoritmos; se investigó y suponemos que es debido al hecho de que se tienen presentes de igual manera los mismos atributos como referencia para la predicción y el porcentaje resultante es similar.

Visto bueno de continuidad

Considerando las evidencias presentadas, se Autoriza, la integración o implementación de los modelos de predicción de género al que se le ha cometido un homicidio (modelo_generoLoR.pickle) y predicción de día en el que se produjo cometido un homicidio (modelo_diaTreeR.pickle) y se Rechaza los modelos de predicción del mes en que se

cometió un homicidio y la predicción de la zona en que se cometió un homicidio puesto que su porcentaje de precisión son menores al 90%.

Autoriza, Elkin Rafael Pulido Farias y Jorge Duban Bernal Cárdenas.

Firma.