

Proyecto fase 2:  
Presentación de resultados

María Fernanda Estrada Cornejo 14198  
Christopher Kevin Sandoval García 13660  
Luis Estuardo Delgado Ordoñez 17187

Minería de Datos



Universidad del Valle de Guatemala  
Facultad de Ingeniería  
Mayo 2020

## I. MÉTODO PARA OBTENER CONJUNTOS DE DATOS

Se utilizó la base de datos de hechos de tránsito del Instituto Nacional de Estadística de Guatemala (<https://www.ine.gob.gt/ine/estadisticas/bases-de-datos/accidentes-de-transito/>). Los archivos tienen formato sav y xls (en algunos años), cada uno correspondiente a un año del 2009-2019. Luego, se procedió a descargar el archivo de hechos de tránsito .sav de cada año. Usando el paquete “haven” en R, se ejecutó el siguiente comando para leer y almacenar el archivo en un data set.

```
accidentes_train<-read_sav("DatosAccidentes/accidentes_2009.sav")
```

Sin embargo, como se debían unir en un solo data frame, se debió hacer limpieza antes (ver sección de transformaciones). Ya con todos los datos juntos, se decidió que el set de datos sería del año 2009-2013, y el año 2014 como test. Se procuró que ambos sets estuvieran balanceados

---

Año:  Período:

---

**-Base de Datos:**

Fallecidos y lesionados	 Abrir SPSS
Hechos de tránsito	 Abrir SPSS
Vehículos involucrados	 Abrir SPSS

---

**-Diccionario de Variables:**

Diccionario de fallecidos y lesionados	 Abrir xls
Diccionario de hechos de tránsito	 Abrir xls
Diccionario de vehículos involucrados	 Abrir xls

## **II. VARIABLE RESPUESTA Y TRANSFORMACIONES**

### **Variable respuesta**

El problema principal del proyecto es determinar si existe una relación entre la condición del conductor al momento del accidente y las características del vehículo que manejaba, así como el día en que ocurrió el accidente y otras características del conductor. Es por esto que nuestra variable respuesta sería la condición del conductor al momento del accidente (normal o bajo efectos de una sustancia).

### **Transformaciones realizadas**

La primera transformación sobre los datos fue cambiar el nombre de las columnas. Los datos originales cambiaban de nombre de variable cada año; por ejemplo, en un año tenían estado\_pil y otro tenían estado\_con. Al establecer cuáles columnas eran iguales, se les cambió al mismo nombre.

Luego, se observó que a partir del año 2015, la variable respuesta fue eliminada. Es decir, la variable "estado\_pil" no se encontraba en los datasets de los años 2015-2019, por lo que no se tomaron en cuenta estos años.

Algunas columnas poseían valores como "9999" o "99", las cuales eran usadas para denotar un "no hay información" o "NA". Estos datos fueron eliminados para que no afectaran el modelo generado.

Para evitar problemas, se normalizaron ciertos datos como fechas, edad, etc (numéricos). Sin embargo, se notó que la cantidad de colores era muy alta, por lo que se dejaron los 9 colores más frecuentes. Estas variables que no son exactamente numéricas (solo utilizadas para representar variables categóricas) no se normalizaron.

Después, se cambiaron los valores de "estado\_pil" de 1-2 a 0-1 para mayor comprensión y que no generara errores. También se estandarizó el formato en el que venían los datos de hora del accidente; por ejemplo, unas iban de 0-23 y otras de 1-24. Se dejó con formato 0-23.

### III. APLICACIÓN DE ALGORITMO

Las variables que se tomaron en cuenta para predecir el estado del conductor fueron día de la semana, hora de ocurrencia, sexo y edad del piloto, tipo y color del vehículo. Se utilizaron los datos transformados para generar modelos de redes neuronales con Caret, redes neuronales con Pcanet y support vector machines. Los modelos son los siguientes, respectivamente:

```
> modeloCaret
Neural Network

5247 samples
  6 predictor
  2 classes: '0', '1'

No pre-processing
Resampling: Bootstrapped (25 reps)
Summary of sample sizes: 5247, 5247, 5247, 5247, 5247, ...
Resampling results across tuning parameters:
```

size	decay	Accuracy	Kappa
1	0e+00	0.5870879	0.1521243
1	1e-04	0.5972358	0.1696290
1	1e-01	0.6029742	0.1715781
3	0e+00	0.6218592	0.2314694
3	1e-04	0.6232986	0.2371078
3	1e-01	0.6495046	0.2814021
5	0e+00	0.6249130	0.2359392
5	1e-04	0.6297215	0.2471321
5	1e-01	0.6515465	0.2856175

Accuracy was used to select the optimal model using the largest value.  
The final values used for the model were size = 5 and decay = 0.1.

```
> modeloCaretPCANNet
Neural Networks with Feature Extraction

5247 samples
  6 predictor
  2 classes: '0', '1'

No pre-processing
Resampling: Bootstrapped (25 reps)
Summary of sample sizes: 5247, 5247, 5247, 5247, 5247, ...
Resampling results across tuning parameters:
```

size	decay	Accuracy	Kappa
1	0e+00	0.5974872	0.1890315
1	1e-04	0.6004846	0.1955654
1	1e-01	0.6053958	0.1906227
3	0e+00	0.6045070	0.2002140
3	1e-04	0.6078887	0.1928804
3	1e-01	0.6141905	0.2081362
5	0e+00	0.6143975	0.2080963
5	1e-04	0.6127904	0.2024479
5	1e-01	0.6343525	0.2476310

Accuracy was used to select the optimal model using the largest value.  
The final values used for the model were size = 5 and decay = 0.1.

```
> modelosvm_1
```

```
Call:
```

```
svm(formula = estado_pil ~ ., data = accidentes_train, cost = 2^5, kernel = "linear")
```

```
Parameters:
```

```
  SVM-Type:  C-classification
```

```
  SVM-Kernel: linear
```

```
    cost:    32
```

```
Number of support Vectors: 4423
```

## IV. PREDICCIÓN Y RESULTADO

### Matriz de confusión

La matriz del modelo de redes neuronales con Caret es la siguiente

```
> cfmCaret
Confusion Matrix and Statistics

      Reference
Prediction 0    1
0  2295  265
1   913  361

      Accuracy : 0.6927
      95% CI   : (0.6779, 0.7073)
No Information Rate : 0.8367
P-Value [Acc > NIR] : 1

      Kappa : 0.2062

McNemar's Test P-Value : <2e-16

      Sensitivity : 0.7154
      Specificity : 0.5767
      Pos Pred Value : 0.8965
      Neg Pred Value : 0.2834
      Prevalence : 0.8367
      Detection Rate : 0.5986
      Detection Prevalence : 0.6677
      Balanced Accuracy : 0.6460
```

La matriz del modelo de redes neuronales con Pcnnet es la siguiente

```
> cfmCaretPCANNet
Confusion Matrix and Statistics

      Reference
Prediction 0    1
0  2237  266
1   971  360

      Accuracy : 0.6774
      95% CI   : (0.6623, 0.6921)
No Information Rate : 0.8367
P-Value [Acc > NIR] : 1

      Kappa : 0.1874

McNemar's Test P-Value : <2e-16

      Sensitivity : 0.6973
      Specificity : 0.5751
      Pos Pred Value : 0.8937
      Neg Pred Value : 0.2705
      Prevalence : 0.8367
      Detection Rate : 0.5835
      Detection Prevalence : 0.6528
      Balanced Accuracy : 0.6362
```

La matriz del modelo de SVM es la siguiente

```
> cfmSVM
Confusion Matrix and Statistics

      Reference
Prediction  0    1
      0 2329  366
      1  879  260

      Accuracy : 0.6753
      95% CI : (0.6602, 0.6901)
      No Information Rate : 0.8367
      P-Value [Acc > NIR] : 1

      Kappa : 0.1063

      Mcnemar's Test P-Value : <2e-16

      Sensitivity : 0.7260
      Specificity : 0.4153
      Pos Pred Value : 0.8642
      Neg Pred Value : 0.2283
      Prevalence : 0.8367
      Detection Rate : 0.6075
      Detection Prevalence : 0.7029
      Balanced Accuracy : 0.5707
```

## Discusión de resultados

Al ejecutar los algoritmos en el set de datos la primera vez, se obtuvo que siempre predecía que el conductor iba bajo efectos de alguna sustancia. Se investigó qué podría ser, ya que el set de training sí contenía datos de conductores sobrios. El problema era que habían demasiados conductores ebrios en el set de datos de training (aprox. 1% contra 99%). Entonces, se procedió a equilibrar mejor entre conductores bajo efectos o no.

Ya con ambos lados equilibrados, se ejecutaron nuevamente los algoritmos y éstos ya lograban predecir ambos casos. En el modelo de redes neuronales usando Caret se obtuvo un accuracy del 69.3%, con más errores al predecir que el conductor iba bajo efectos de una sustancia cuando no era así. En el modelo de redes neuronales usando Pccanet se obtuvo un accuracy del 67.8%, igualmente con más errores al predecir que el conductor iba bajo efectos de una sustancia cuando no era así. En el modelo de SVM se obtuvo un accuracy del 67.5%, igualmente con más errores al predecir que el conductor iba bajo efectos de una sustancia cuando no era así.

Analizando los tres casos en general, el mayor error que tuvieron fue clasificar el estado del conductor como bajo efectos de una sustancia cuando no era así. Aunque es un error grave, es mejor aclararlo que no notar que sí lo está y dejarlo ir cuando podría ocasionar otro accidente. En conclusión el mejor modelo fue el de redes neuronales utilizando Caret, ya que tuvo el mejor accuracy y fue el más rápido en ejecutar.