

Benemérita Universidad Autónoma de Puebla - FCC

Minería de Datos

Avance IV.

Validación Cruzada



SEMANA DEL 11 - 15 DE MAYO 2020

Emiliano Carrillo Moncayo

[Enlace al Jupyter Notebook de esta entrega](#)

Introducción y Objetivos

En mis entregas anteriores había ocupado el método de train/test split para evitar el overfitting y underfitting lo cual no había sido lo más óptimo. Por lo que en esta entrega decidí implementar el método de k-fold Validación Cruzada para partir mi set de datos en tantos pliegues como lo indique el usuario en el input. Tras haber hecho esto observé que la precisión de mis clasificadores y de mi ensamblador final mejoraba.

Además de esto, comencé a organizar mi código para la entrega final. Ahora le proporciono la habilidad al usuario de introducir los parámetros de cada modelo que ocupo, de mi PCA y del ensamble pues esto no lo hacía con anterioridad.

Experimentos y Desarrollo

Al aplicar la validación cruzada muestro la puntuación que se obtuvo en cada pliegue y la precisión promedio del proceso con su respectiva desviación.

Además para cada uno de los modelos gráfico la gráfica de dispersión comparando los datos que predijo en contra de los verdaderos.

```
In [199]: from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import cross_val_score, cross_val_predict
from sklearn import metrics

model_1 = KNeighborsClassifier(n_neighbors= neighborsWidget.value)

scores = cross_val_score(model_1, x_fin, y_fin, cv=folds)

print ("Puntuaciones de la Validación Cruzada:\n", scores)
print ("\nPrecisión del k-Fold:\n %.2f (+/- %.2f)" % (scores.mean(), scores.std() * 2))

Puntuaciones de la Validación Cruzada:
[0.79591837 0.97959184 0.95918367 0.97959184 0.97916667 0.9375
 0.97916667 1.         0.91666667 0.91666667]

Precisión del k-Fold:
0.94 (+/- 0.11)
```

```
In [204]: pred_1 = cross_val_predict(model_1, x_fin, y_fin, cv= folds)
plt.scatter(y_fin, pred_1)
```

Out[204]: <matplotlib.collections.PathCollection at 0x1a271b1b90>



RESULTADOS DEL CLASIFICADOR KNN TRAS APLICARLE VALIDACIÓN CRUZADA CON K=10

Como dije, también comencé a poner inputs para que el usuario tenga la posibilidad de experimentar con todos los parámetros de mi ensamble y de los modelos que lo componen.

kFolds:	<input type="text" value="10"/>
vecinos:	<input type="text" value="5"/>

EJEMPLO DE INPUTS DEL MODELO KNN

Siguientes Pasos

Para la siguiente entrega planeo:

1. Seguir optimizando y compactando el código
2. Encontrar una mejor manera de graficar los resultados predicción contra valores reales para cada modelo y el ensamble general.
3. Mejorar la Interfaz de Usuario y colocar todos los inputs en un solo lugar para que la experiencia de éste sea óptima al ejecutar el programa.
4. Graficar los resultados del ensamble