Selección de variables

Máster en Data Science

Mario Encinar, PhD MAPFRE
encinar@ucm.es

La **selección de variables** en aprendizaje automático (supervisado) es el proceso destinado a construir y elegir subconjuntos de variables especialmente útiles para la construcción de un buen modelo predictivo.

Fundamentalmente, se persiguen cuatro objetivos:

- Facilitar el entendimiento y la visualización de los datos.
- Reducir los requisitos de almacenamiento.
- Reducir los tiempos de entrenamiento y utilización de los modelos.
- Desafiar la *curse of dimensionality* para mejorar la calidad de las predicciones.

Véase The Curse of Dimensionality in classification y If you have high dimensional data, then almost all the data are outliers

Para ver (y tirar del hilo):

- What Makes a Good Feature? - Machine Learning Recipes #3
- Intro to Feature Engineering with TensorFlow - Machine Learning Recipes #9
- What is Machine Learning?
- The 7 Steps of Machine Learning

Checklist:

(1) **¿Tienes conocimiento de negocio?** En caso afirmativo, construye y considera las variables relevantes para el problema.

Andrew Ng: *Coming up with new features is difficult, time-consuming, requires expert knowledge. "Applied machine learning" is basically feature engineering*.

Referencias:

Barzilay, O., Brailovsky, V.L., *On domain knowledge and feature selection using a support vector machine*. Disponible aquí.
Groves, W., *Using Domain Knowledge to Systematically Guide Feature Selection*. Disponible aquí.

Checklist:

(2) **¿Están tus variables en la misma escala?** En caso negativo, considera la posibilidad de normalizarlas.

- Los modelos más sencillos ganan en explicabilidad.
- Los algoritmos basados en distancias dan, naturalmente, más peso a variables con escalas *grandes*, induciendo posibles errores.
- Algunos algoritmos (SVMs, fundamentalmente) reducen su tiempo de entrenamiento cuando todas las variables están en la misma escala.

Checklist:

(3) **¿Necesitas reducir el número de variables?**

- Por coste o tiempo.
- Por interpretabilidad

En caso afirmativo:

- Descarta variables a partir del análisis exploratorio.
- Utiliza un método de filtrado *a priori* para descartar variables.
- Construye modelos *sencillos* y con pocas variables que te den intuición acerca de la fuerza predictiva de las variables, y descarta las peores.

Checklist:

(4) **Utiliza algún método de *rankeo* de variables**.

- Para evaluar las variables individualmente.
- Para conseguir un *benchmark* del modelo que construyas más adelante.
- Para eliminar (o, más generalmente, tratar) *outliers*.

Checklist:

(5) **¿Tienes limitaciones de tiempo?** Si es que no, prueba todo lo que se te ocurra. En caso contrario:

- Comienza por un modelo sencillo (¿lineal?) empleando las variables de *ranking* mayor.
- Compara su rendimiento con un modelo del mismo tipo utilizando un método de selección *stepwise*.
- Construye modelos más complicados empleando variables sugeridas por el proceso anterior.
- ...

Dado un problema de aprendizaje supervisado supervisado del tipo "predecir la respuesta Y en base a los predictores $X_1, \ldots, X_p$", el *rankeo* de variables consiste en

- Considerar una función de *score* S de tal modo que, para cada $j = 1, \ldots, p$, $S(j)$ represente el *valor* de la variable X_j a la hora de predecir Y.
- Ordenar las variables en sentido decreciente de S.
- Utilizar las p' (a determinar) variables mejor colocadas en el *ranking* para construir el modelo.

Los distintos métodos de filtrado se distinguen entre sí por la función de *score* y por cómo se elige p' (normalmente, por un criterio *del codo*).

Algunas funciones de *score*:

- En problemas de regresión:
  - $S(j) = \rho(X_j, Y)^2$.

    donde ρ puede ser un coeficiente de correlación (Pearson's, Spearman's), entre otros.
  - $S(j) = \max\{\rho(X_j, Y)^2, \rho(\log(X_j), Y)^2, \rho(X_j^2, Y)^2\}$.
  - $S(j) = R^2_{Y|X_j}$, para algún modelo $Y \sim X_j$ (equivalentemente η en un one-way ANOVA).
- En problemas de clasificación:
  - $S(j) = \rho(X_j, Y)^2$.

    donde ρ puede ser un coeficiente de correlación (Spearman's), χ^2, un estadístico F (one-way ANOVA), entre otros.
  - $S(j) = \kappa_{Y|X_j}$, para algún modelo $Y \sim X_j$.

    donde κ es una medida de Observed accuracy/Predicted Accuracy.

- En general, tanto para regresión como para clasificación, sirve la *información mutua*:

$$I(X_j, Y) = \iint f(x_j, y) \log \left(\frac{f(x_j, y)}{f_{X_j}(x_j) f_Y(y)} \right) dx_j dy$$

para variables continuas,

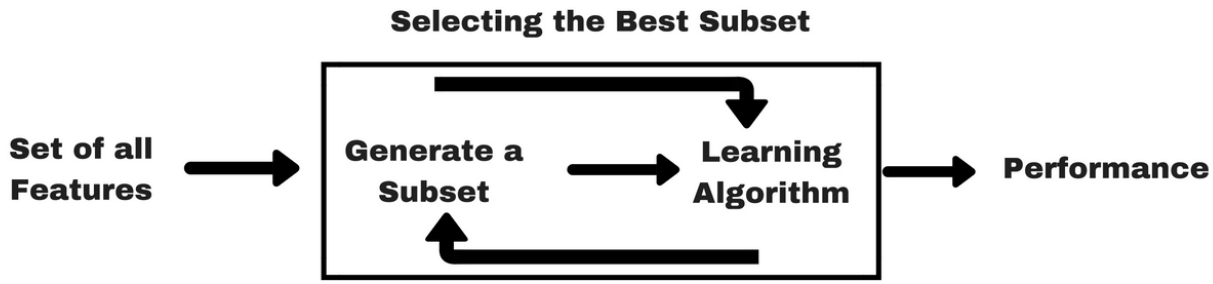
$$I(X_j, Y) = \sum_{y \in Y} \sum_{x_j \in X_j} p(x_j, y) \log \left(\frac{p(x_j, y)}{p(x_j)\, p(y)} \right)$$

para variables discretas.
Si tienes una variable continua y otra discreta, puedes discretizar la variable continua.

El problema fundamental que presentan estos métodos de filtrado es que pueden eliminar variables que, por sí solas, sean *malas*, pero que sean útiles en combinación con otras.

A	B	A **XOR** B
0	0	0
0	1	1
1	0	1
1	1	0

Añadido al problema de elegir una discretización adecuada de las variables continuas, paso necesario para utilizar los estadísticos χ^2, correlación de Spearman o la Información mutua.



Los métodos *wrapper* utilizan un algoritmo de aprendizaje automático como una *caja negra* para seleccionar variables:

- Se considera una cierta familia $\mathcal{F}$ de subconjuntos del total de predictores.
- Para cada $S \in \mathcal{F}$, se entrena un modelo M_S de predicción (con el algoritmo fijado) utilizando S como conjunto de predictores.
- Se elige el mejor conjunto de variables como aquel S que maximiza el rendimiento (por ejemplo, en *cross-validation* de M_S).

Los **métodos *stepwise*** son el caso más habitual de *wrappers* que se emplean en la práctica, aunque existen otros casos de uso frecuente como el *random-hill climbing algorithm* o la *recursive feature ellimination*.

Random-hill climbing algorithm: Es una modificación de los métodos *stepwise* "tradicionales": en cada paso, en lugar de elegir *la mejor* variable que añadir o quitar, se elige una de manera aleatoria (con una distribución de probabilidad adecuada).

Recursive feature ellimination: Es una modificación de la *backward stepwise selection*: en cada iteración, nos quedamos con los predictores de mayor importancia predictiva.

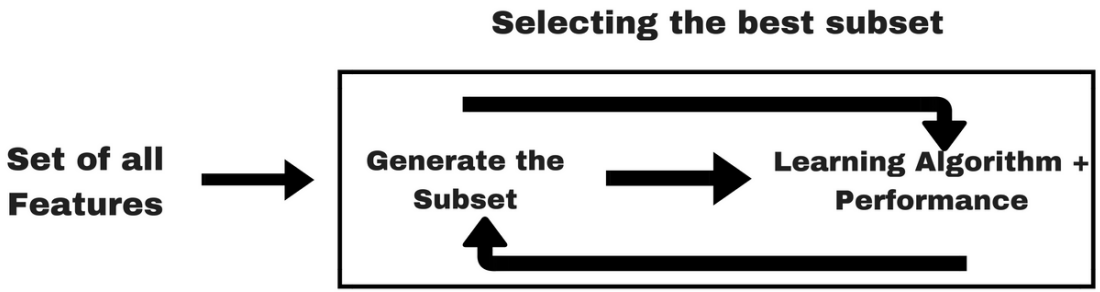
Referencias:

Kohavi, R., John, G.H., *Wrappers for feature subset selection*. Disponible aquí.

Skalak, D., *Prototype and Feature Selection by Random Mutation Hill Climbing Algorithms*. Disponible aquí.

Paquetes interesantes para la Selección de Variables (Filters, Wrappers):

- Selección de variables con caret.
- Selección de variables con scikit-learn.
- R Fselector package.
- IBM Python package ibmdbpy
- R leaps package for linear regression.



Los métodos *embedded* son aquellos que realizan la selección de variables como parte del proceso de entrenamiento de algún modelo.

Los ejemplos de referencia deben ser el árbol de decisión (con *pruning*) y la regresión Lasso, donde la selección de variables se implementa para evitar el *overfitting*.

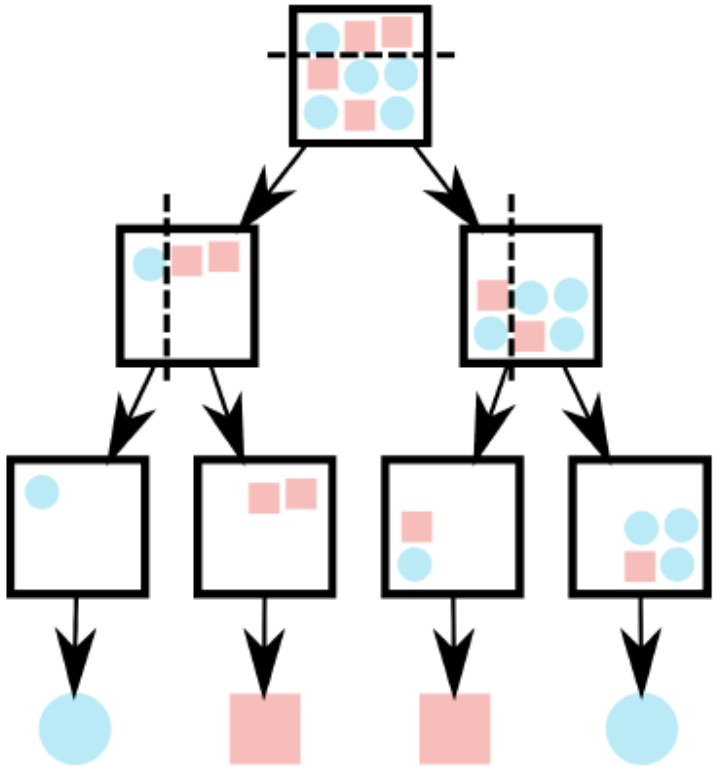
Regresión Lasso: En el marco de un problema de regresión lineal, en lugar de hallar β que minimice

$$\sum_{i=1}^{n} (y_i - \beta^t x_i)^2,$$

se halla β que minimice

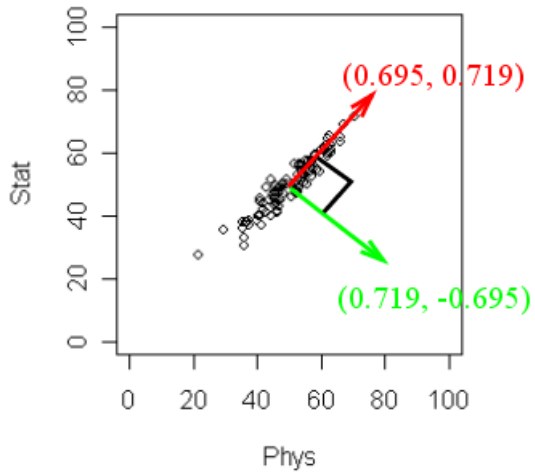
$$\sum_{i=1}^{n} (y_i - \beta^t x_i)^2 + \lambda \sum_{j=1}^{p} |\beta_j|.$$

Árbol de decisión:



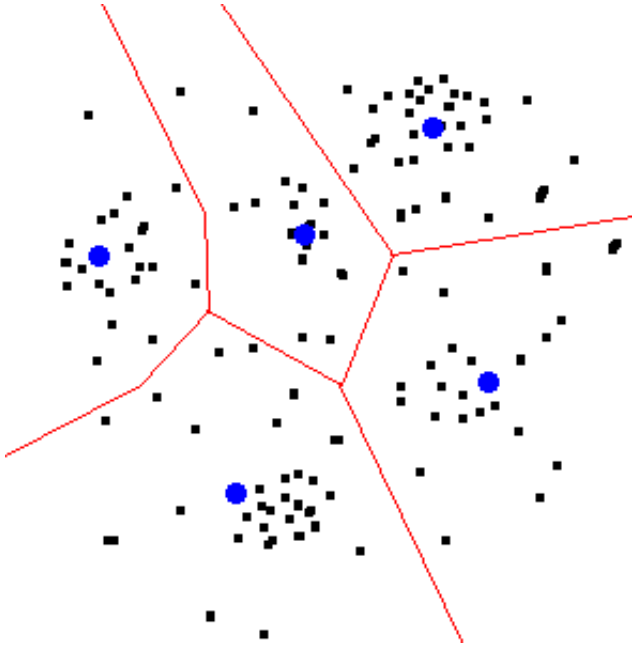
Cuando hay demasiadas variables o cuando el significado de las variables no es claro y las técnicas anteriores no dan buenos resultados, suelen emplearse técnicas de reducción de dimensiones para seleccionar variables.

La técnica más habitual en este sentido es el **análisis de componentes principales**.



El problema fundamental de esta aproximación es que sólo se tienen en cuenta relaciones lineales entre las variables originales... Pero se puede arreglar.

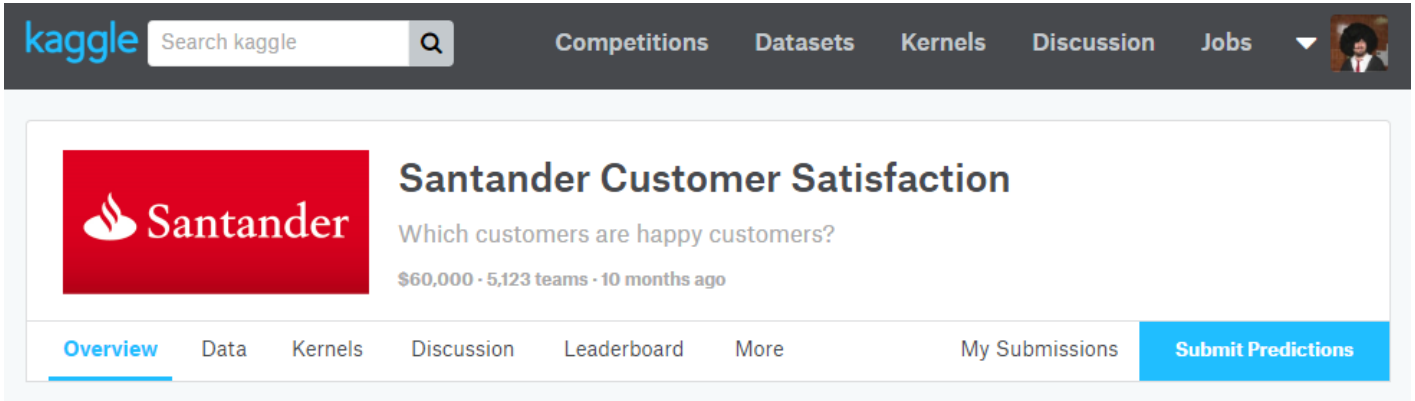
Otra posibilidad es aplicar **clustering** en el espacio de las variables y sustituir cada cluster de variables por un representante *apropiado* del cluster.



Metodología *informal*:

- Hay que tratar de evitar usar todas las variables en el entrenamiento de un modelo (salvo que haya tiempo y máquina *ilimitados*).
- Entrenar modelos con pocas variables para descartar las malas.
- Llegar a una hipótesis básica con pocas variables.
- Hacer *forward stepwise*.
- Remuestrear y repetir.

Tampoco hay que volverse locos...



Clasificación. 76020 filas, 370 predictores.

Una solución aquí:

1. Eliminación de variables constantes.
2. Eliminación de columnas duplicadas.
3. XGBoost.

ROC = 0,825; ROC del ganador = 0,829.

Ejemplos:

- How important is feature selection? (Kaggle)
- Feature Selection and Data Visualization (Kaggle)

Referencias:

- Guyon, I., Elisseef, A., *An Introduction to Variable and Feature Selection*, Journal of Machine Learning Research 3 (2003), pp. 1157-1182.