Feature Selection and Engineering

May 17, 2017

- Difference between population and sample
- Define covariance and standard deviation (sample approximation)

1 Feature Engineering

SVD based

- based on finite SVD
- linear feature construction
- frequently used in recos (matrix factorization)

Autoencoder

- Non linear extenstion of SVD via NN
- could be also used in a reco

2 Feature Selection

Missing Values Ratio

Data columns with too many missing values are unlikely to carry much useful information. Thus data columns with number of missing values greater than a given threshold can be removed. The higher the threshold, the more aggressive the reduction.

Low Variance Filter

Similarly to the previous technique, data columns with little changes in the data carry little information. Thus all data columns with variance lower than a given threshold are removed. A word of caution: variance is range dependent; therefore normalization is required before applying this technique.

High Correlation Filter

Data columns with very similar trends are also likely to carry very similar information. In this case, only one of them will suffice to feed the machine learning model. Here we calculate the correlation coefficient between numerical columns and between nominal columns as the *Pearsons Product Moment Coefficient* and the *Pearson's chi square value* respectively. Pairs of columns with correlation coefficient higher than a threshold are reduced to only one. A word of caution: correlation is scale sensitive; therefore column normalization is required for a meaningful correlation comparison.

Definition 1 (Correlation)**.**

Definition 2 (Person product moment coefficient)**.** *Is also known as correlation coefficient or r coefficient. For a population*

$$r_{XY} = \frac{cov(X,Y)}{\sigma_x \sigma_y} \tag{1}$$

with covarince cov and standard deviation σ.

Now we consider this definition for a sample. Let $(x_i, y_i),\ i = 1 \dots n$ be a random sample with sample mean $\bar{x}, \bar{y}$ (i.e., $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$). The correlation coefficient assumes then the form[1]

$$r_{xy} = \frac{(x-\bar{x})\,(y-\bar{y})}{\|x-\bar{x}\|_2 \|y-\bar{y}\|_2} \tag{2}$$

So, this is nothing than a normalized scalar product of feature vectors shifted by their sample mean. Note that this is the sample definition, whereas there is another definition for the population.

Definition 3 (Pearson chi square value)**.**

Random Forests / Ensemble Trees

Decision Tree Ensembles, also referred to as random forests, are useful for feature selection in addition to being effective classifiers. One approach to dimensionality reduction is to generate a large and carefully constructed set of trees against a target attribute and then use each attributes usage statistics to find the most informative subset of features. Specifically, we can generate a large set (2000) of very shallow trees (2 levels), with each tree being trained on a small fraction (3) of the total number of attributes. If an attribute is often selected as best split, it is most likely an informative feature to retain. A score calculated on the attribute usage statistics in the random forest tells us –relative to the other attributes– which are the most predictive attributes.

Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a statistical procedure that orthogonally transforms the original n coordinates of a data set into a new set of n

[1] Lack of knowledge here: when are the sample approximations good - for certain distributions? See lecture notes Gesine Reiner, *Statistical Theory*

coordinates called principal components. As a result of the transformation, the first principal component has the largest possible variance; each succeeding component has the highest possible variance under the constraint that it is orthogonal to (i.e., uncorrelated with) the preceding components. Keeping only the first m ¡ n components reduces the data dimensionality while retaining most of the data information, i.e. the variation in the data. Notice that the PCA transformation is sensitive to the relative scaling of the original variables. Data column ranges need to be normalized before applying PCA. Also notice that the new coordinates (PCs) are not real system-produced variables anymore. Applying PCA to your data set loses its interpretability. If interpretability of the results is important for your analysis, PCA is not the transformation for your project.

Backward Feature Elimination

In this technique, at a given iteration, the selected classification algorithm is trained on n input features. Then we remove one input feature at a time and train the same model on $n-1$ input features n times. The input feature whose removal has produced the smallest increase in the error rate is removed, leaving us with $n-1$ input features. The classification is then repeated using $n-2$ features, and so on. Each iteration k produces a model trained on $n-k$ features and an error rate $e(k)$. Selecting the maximum tolerable error rate, we define the smallest number of features necessary to reach that classification performance with the selected machine learning algorithm.

Forward Feature Construction

This is the inverse process to the Backward Feature Elimination. We start with 1 feature only, progressively adding 1 feature at a time, i.e. the feature that produces the highest increase in performance. Both algorithms, Backward Feature Elimination and Forward Feature Construction, are quite time and computationally expensive. They are practically only applicable to a data set with an already relatively low number of input columns.

Shannon Entropy

The Shannon Entropy has the advantage over correlation-based approaches because it also captures non-linear effects. The basic idea is to include those features that maximise the target (label) entropy $H(L)$ and the conditional entropy of the label entropy given feature ν

$$\mathrm{argmax}_{v}\left(H(L) - H(L|\nu)\right) \tag{3}$$

The Term

$$I(L;\nu) := (H(L) - H(L|\nu)) \tag{4}$$

is also called mutual information. Thus, it will pick those features whose conditional entropy is minimal, which implies the maximally skewed feature with respect to the target.

The details follow now:

Definition 4 (Shannon Entropy)**.** *Let there be given a discrete random variable X with possible values $x_1 \dots x_n$ and probability mass function $P(X)$. The Shannon Entropy is defined as the expectation value of the information content of X $I(X) := -\ln P(X)$,*

$$H(X) = E[-\ln P(X)] \tag{5}$$

For a finite sample S this reads,

$$H(X) = -\sum_i P(X = x_i) \ln P(X = x_i) \tag{6}$$

Properties of the entropy

- $\lim_{x\to 0} x \ln x = 0$ as Pablo has shown. Rewrite $x \ln x = \frac{\ln x}{x^{-1}}$ and use L'Hospital.

- The entropy is minimized for the maximally skewed distribution. i.e., a probability mass function of the form $(\{0, \dots 0, 1, 0 \dots 0\})$. For probabilities $\{p_k \geq 0 | \sum p_k = 1\}$ the term $-p_k \ln p_k \geq 0$ where it takes the value zero for $p_k = 0, 1$. Therefore the minimum $H(X) = 0$ is achieved for the maximally skewed distribution.

- The Entropy is maximised for the uniform distribution.

  Proof. To see this maximise the entropy under the side condition that the probabilities add up to one – i.e., the Lagrange function assumes the form

  $$L(p_k) = -\sum_{k=1}^{N} p_k \ln p_k - \lambda \left(\sum_{k=1}^{N} p_k - 1 \right) \tag{7}$$

  Taking the derivatives gives the set of equations

  $$0 = -(\ln p_i + 1) - \lambda \tag{8}$$

  $$0 = \sum_{k=1}^{N} p_k - 1. \tag{9}$$

  From Eq. (8) $\Rightarrow p_i = e^{-(1+\lambda)}$. Substituting this result in Eq. (9) we get $e^{-(1+\lambda)} = \frac{1}{N}$, which eventually gives $p_i = 1/N \ \forall i$. □

The Entropy of a (discrete) random variable X conditioned to a random variable Y taking the value y is then given by

$$H(X|y) = E_{X|y}[-\ln P(X|y)] \tag{10}$$

The conditional Entropy is then given by the expectation value with respect to Y.

Definition 5 (Conditional Entropy)**.**

$$H(X|Y) = E_Y[E_{X|y}[-\ln P(X|Y)]] \tag{11}$$

Taking samples, we evaluate the conditional entropy to a working equation,

Theorem 1. *The conditional entropy for two events X, Y taking values x_i and y_i, respectively is given by*

$$H(X|Y) = -\sum_{i,j} P(x_i, y_j) \ln \frac{P(x_i, y_j)}{P(y_j)} \tag{12}$$

Proof. According to definition 5 we have

$$\begin{aligned} H(X|Y) &= E_Y[E_X|y[-\ln P(X|Y)]] \\ &= -\sum_j P(y_j) \sum_i P(x_i|y_j) \ln P(x_i|y_j) \\ &= -\sum_{i,j} P(x_i, y_j) \ln \frac{P(x_i, y_j)}{P(y_j)} \end{aligned}$$

□

Properties of the conditional entropy

- $H(X|Y) \leq H(X)$
  However, an individual y_i $H(X|Y = y_i)$ may exceed $H(X)$
- **For independent X and Y:** $H(X|Y) = H(X)$
- **For X depending on Y:** $H(X|Y) < H(X)$. This is exactly what we are exploiting for feature selection. From the set of possible features we choose feature ν, for which $H(X|\nu)$ is as small as possible. This means the conditional Probability mass function $P(X|\nu)$ is maximally skewed.
- **Chain rule for the conditional entropy**
  $H(X|Y) = H(X, Y) - H(Y)$. Follows immediately from Theorem 1.
- **Bayes rule for the conditional entropy**
  $H(X|Y) = H(Y|X) - H(Y) + H(X)$. Follows from the Chain rule above.

The Shannon Entropy is restricted to discrete values. The continuous version is called differential entropy.

Definition 6 (Differential entropy)**.**

$$h(X) = -\int_{\mathbb{X}} p(x) \ln p(x) dx \tag{13}$$

Properties of the mutual information

- Symmetric
- Can me tweaked to fulfill metric properties:

$$\begin{aligned} d(X, Y) &= H(X, Y) - I(X; Y) \\ &= H(X) + H(Y) - 2I(X; Y) \\ &= H(X|Y) + H(Y|X) \end{aligned}$$

Neural Networks

Gini index