Regularized Linear Regression

April 1st, 2019

Regularization

Complex learning models may lead to unstable behavior

- Complex learning algorithms can become **unstable**; i.e., highly dependent on the training data
- Instability is a manifestation of a tendency of overfitting
- ***Regularization*** is a general method to avoid such overfitting by applying additional constraints to the weight vector
- A common strategy is to make sure that the weight are, on average, small in magnitude, which is known as ***shrinkage***

Unstable learning algorithm tends to overfit

- A regularization function measures the complexity of the hypotheses
- It can be also seen as a ***stabilizer*** of the learning algorithm
- An algorithm is considered ***stable*** if a ***slight change*** of its input ***does not change*** its ***output much***.
- Let A be a learning algorithm, $S = (z_i, \ldots, z_m)$ be a training set of m examples and $A(S)$ denote the output of A
- We can say that algorithm A is suffering from overfitting if the difference between the true risk of its output $L_d(A(S))$, and the empirical risk of its output $L_s(A(S))$ is large.
- Thus, our interest is in the expectation

$$\mathbb{E}_s[L_{\mathfrak{D}}(A(S)) - L_s(A(S))$$

Unstable learning algorithm tends to overfit

- In this case, stability can be defines as: let z' be an additional example and $S^{(i)}$ be the training set obtained by replacing the i^{th} example of S, $S^{(i)} = (z_i, \ldots, z_{i-1}, z', z_{i+1}, \ldots, z_m)$
- Thus, stability measures the effect of the small change of the input on the output of A by comparing the loss of the hypotheses $A(S)$ on z_i to the loss of the hypotheses $A(S(^{(i)}))$ on z_i.
- Consequently, a good learning algorithm will have $\ell(A(S^{(i)}), z_i) - \ell(A(S), z_i) \geq 0$, since in the first term the learning algorithm does not observe the example z_i while in the second the term z_i is indeed observed. If the difference is very large, the learning algorithm might been overfitting

Ridge Regression

Ridge Regression

- It is based on sum of squared residuals penalty

$$\hat{\beta}_{ridge} = \arg\min_{\beta} (y - X\beta)^T (y - X\beta) + \lambda ||\beta||^2$$

- where $||\beta||^2 = \sum_{i=1}^{p} \beta_i^2$ is the squared norm of the vector β, or equivalently the dot product $\beta^T \beta$
- λ is a scalar determining the amount of the regularization
- Its closed-form can be written as:

$$\hat{\beta} = (X^T X + \lambda I)^{-1} X^T y$$

- Ridge regression shrinks the coefficients towards 0, but does not lead to a ***sparse model***

Lasso

Lasso

$$\hat{\beta}_{lasso} = \arg\min_{\beta} ||y - \beta||_2^2 + \lambda||\beta||_1$$

- It stands for *Least absolute shrinkage and selection operator*
- It replaces the ridge regularization term $\sum_{i=1}^{p} \beta_i^2$ with the sum of the absolute weights $\sum_{i=1}^{p} |\beta_i|$
- Thus, lasso uses L_1 regularization, whereas ridge regression uses the L_2 norm
- Lasso regression favors sparse solutions

Lasso

- It is quite sensitive to the regularization parameter λ, which is usually set on hold-out data or in cross-validation
- Therefore, there is no closed form solution and numerical optimization technique must be applied.

In summary...

- Ridge regression
  - correlated variables get similar weights
  - identical variables get identical weights
  - It is not sparse
- Lasso
  - correlated variables are randomly picked out
  - It is sparse

References

- Hal Daume III. *A Course in Machine Learning*. 2nd. Self-published, 2017. URL: http://ciml.info/dl/v0_99/ciml-v0_99-all.pdf
  1. **Regularization**: sessions 7.2 and 7.3
- Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd. Springer, 2016. URL: https://web.stanford.edu/~hastie/Papers/ESLII.pdf
  1. **Regularization**: session 10.12
  2. **Ridge regression**: session 3.4.1
  3. **Lasso**: session 3.4.2