Regression and extensions

Victor Kitov

v.v.kitov@yandex.ru

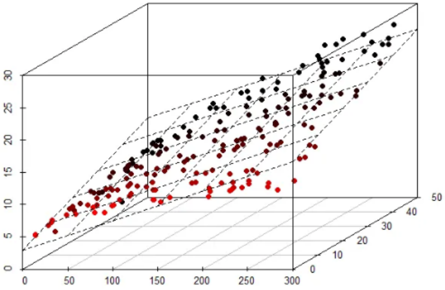


Table of Contents

Linear regression

- Linear model

$$\widehat{y} = x^T \widehat{\beta} = \sum_{i=1}^{D} \widehat{\beta}_i x^i$$

$$\widehat{\beta} = \arg\min_{\beta} \sum_{n=1}^{N} \left(x_n^T \beta - y_n \right)^2$$

- If β_0 is not specified explicitly, include constant feature in x
- Assumptions:
  - each x^i has linear impact with weight β_i on y
  - impact of x^i does not depend on other features.

Method analysis

Advantages:

- interpretability
  - sign of coefficients=direction of influence of x^i
  - modulus of coefficient=strength of influence of x^i (with features from the same scale!)
  - $\widehat{\beta}$ are asymptotically normal (see link), we can test:
    - the significance of the difference between a coefficient and zero (or a group of coefficients from zero)
    - the hypothesis of the positive influence of the feature on the response (positiveness of the coefficient)
- there is an analytical solution
- forecasts are made quickly and easily
- less overfitting compared to complex models
  - for large D can be an optimal model

Disadvantages: model assumptions are too simple

- signs can influence non-linearly
- signs can have interdependent influence

Features

- You can use real and binary features.
- Categorical features can be encoded using:
  - category number (bad)
  - category occurrence counter
  - one-hot encoding (binary)
  - mean value encoding (real)

One-hot encoding

Row Number	Direction
1	North
2	North-West
3	South
4	East
5	North-West

$\Rightarrow$

Row Number	Direction_N	Direction_S	Direction_W	Direction_E	Direction_NW
1	1	0	0	0	0
2	0	0	0	0	1
3	0	1	0	0	0
4	0	0	0	1	0
5	0	0	0	0	1

Mean value encoding

- feature value -> average y, given that feature value:

id	job	job_mean	target
1	Doctor	0,50	1
2	Doctor	0,50	0
3	Doctor	0,50	1
4	Doctor	0,50	0
5	Teacher	1	1
6	Teacher	1	1
7	Engineer	0,50	0
8	Engineer	0,50	1
9	Waiter	1	1
10	Driver	0	0

- Use separate training set for averaging target.
- Also may substitute with average value of another feature.

Solution

Define $X \in \mathbb{R}^{NxD}$, $\{X\}_{ij}$ defines the j-th feature of i-th object, $Y \in \mathbb{R}^n$, $\{Y\}_i$ - target value for i-th object.
Ordinary least squares (OLS) method:

$$L(\beta) = \sum_{n=1}^{N} \left(x_n^T \beta - y_n \right)^2 = \|X\beta - Y\|_2^2 \to \min_\beta$$

$$L'(\beta) = 2\sum_{n=1}^{N} x_n \left(x_n^T \beta - y_n \right) = 0$$

In matrix form:

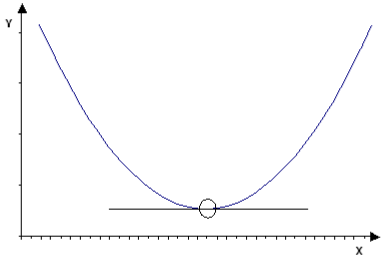
$$2X^T(X\beta - Y) = 0$$

$$\widehat{\beta} = (X^T X)^{-1} X^T Y$$

Intuition: β_i is proportional to covariance between x_n^i and y_n, normalized by $Var[x^i]$ and $cov[x^i, x^j]$.

Comments

- This is the global minimum, because the optimized criteria is convex.
    - convex function of linear function is convex[1]
    - sum of convex functions is convex
    - for convex function the sufficient condition of global minimum is zero gradient:



[1] Will superposition of two convex functions be convex?

Linearly dependent features

- Solution $\widehat{\beta} = (X^T X)^{-1} X^T Y$ exists when $X^T X$ is non-degenerate.
- Problem occurs when one of the features is a linear combination of the other.
  - because of the property $\forall X : \, rank(X) = rank(X^T X)$

Linearly dependent features

- Solution $\widehat{\beta} = (X^T X)^{-1} X^T Y$ exists when $X^T X$ is non-degenerate.
- Problem occurs when one of the features is a linear combination of the other.
  - because of the property $\forall X: \ rank(X) = rank(X^T X)$
  - example: constant unity feature c and one-hot-encoding $e_1, e_2, ...e_K$, because $\sum_k e_k \equiv c$
  - interpretation: non-identifiability of $\widehat{\beta}$ for linearly dependent features:
    - linear dependence: $\exists \alpha: \ x^T \alpha = 0 \, \forall x$
    - suppose β solves linear regression $y = x^T \beta$
    - then $x^T \beta \equiv x^T \beta + k x^T \alpha \equiv x^T (\beta + k\alpha)$, so $\beta + k\alpha$ is also a solution!

Linearly dependent features

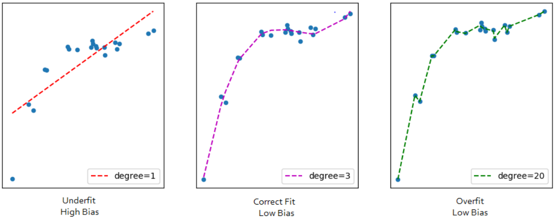
- Problem may be solved by:
    - feature selection
    - dimensionality reduction
    - imposing additional requirements on the solution (regularization)
        - e.g. $\|\beta\|$ should be small

Generalization by nonlinear transformations

Transform $x \in \mathbb{R}^D$ using non-linear transformation $\in \mathbb{R}^M$:
Nonlinearity by x in linear regression may be achieved by applying non-linear transformations to the features:

$$x \to [\phi_1(x), \phi_2(x), \dots \phi_M(x)]$$

$$\widehat{y}(x) = \phi(x)^T \widehat{\beta} = \sum_{m=1}^{M} \widehat{\beta}_m \phi_m(x)$$



Regression with polynomial feature tranformation.

Analysis

The model remains linear in β, so all advantages of linear regression remain:

- interpretability
- closed form solution
- global optimum

Typical transformations

Consider typical feature transformations:

$\phi_k(x)$	**motivation examples**
$(x^i)^2, \sqrt{x^i}$, $\ln x^i$	we take into account the non-linear influence of the distance to the metro on the cost of an apartment
$\mathbb{I}\left\{x^i \in [a, b]\right\}$	Does the client belong to a certain age? (adult, but not retired)
$x^i \mathbb{I}[x^i \leq a],\ \ x^i \mathbb{I}[x^i > a]$	change of impact of x^i after $x^i > a$
$(x^i)\ (x^j)$	width x height = square
$\langle x, z\rangle/(\lVert x\rVert\, \lVert z\rVert)$	angle between object and representative object z
$\lVert x - z\rVert^2$	distance (may use similarity) from object to representative object z
x^i/x^j	flat price/square = cost per meter
$F_{x^i}(x^{\mathrm{i}})$	make feature distribution uniform ($F(\cdot)$ - distribution function)

Non-linear regression

- Alternatively we can model $\mathcal{X} \to \mathcal{Y}$ with arbitrary non-linear function $\widehat{y} = f(x|\theta)$

$$L(\theta|X,Y) = \sum_{n=1}^{N} \left(f(x_n|\theta) - y_n\right)^2$$

$$\widehat{\theta} = \arg\min_{\theta} L(\theta|X,Y)$$

- No analytical solution for $\widehat{\theta}$ will exist in general
    - need numeric optimization methods.

Table of Contents

Regularization

- Overfitting problem: not only *accuracy* matters for the solution but also *model simplicity*!
- Estimate model complexity with regularizer $R(\beta)$:

$$L(\beta) + \lambda R(\beta) = \sum_{n=1}^{N} \left(x_n^T \beta - y_n \right)^2 + \lambda R(\beta) \to \min_{\beta}$$

- $\lambda > 0$ - hyperparameter (how simple model we want).

$$\begin{array}{ll} R(\beta) = ||\beta||_1, & \text{Lasso regression} \\ R(\beta) = ||\beta||_2^2 & \text{Ridge regression} \end{array}$$

- λ controls complexity of the model:

Regularization

- Overfitting problem: not only *accuracy* matters for the solution but also *model simplicity*!
- Estimate model complexity with regularizer $R(\beta)$:

$$L(\beta)+\lambda R(\beta)=\sum_{n=1}^{N}\left(x_n^T\beta-y_n\right)^2+\lambda R(\beta)\to\min_{\beta}$$

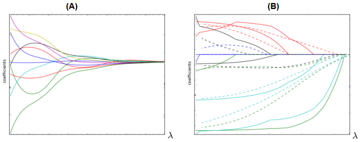
- $\lambda>0$ - hyperparameter (how simple model we want).

$$\begin{array}{ll} R(\beta)=||\beta||_1, & \text{Lasso regression} \\ R(\beta)=||\beta||_2^2 & \text{Ridge regression} \end{array}$$

- λ controls complexity of the model:$\uparrow\lambda\Leftrightarrow$ complexity$\downarrow$.

Comments

- Dependency of β from λ for ridge (A) and LASSO (B):



- LASSO can be used for automatic feature selection.
- λ is usually found using cross-validation on exponential grid, e.g. $[10^{-6}, 10^{-5}, ...10^{5}, 10^{6}]$.
- It's always recommended to use regularization because
  - it gives smooth control over model complexity.
  - removes ambiguity for multiple solutions case.

ElasticNet

- ElasticNet:

$$R(\beta) = \alpha||\beta||_1 + (1-\alpha)||\beta||_2^2 \to \min_\beta$$

$\alpha \in (0,1)$ - hyperparameter, controlling impact of each part.

- If two features x^i and x^j are equal:
    - LASSO may take only one of them
    - ridge will take both with equal weight
        - but it doesn't remove useless features
    - ElasticNet both removes useless features but gives equal weight for usefull equal features
        - better, because we have no reasons to prefer one feature over another

Ridge regression solution

Ridge regression criterion

$$\sum_{n=1}^{N}\left(x_n^T\beta-y_n\right)^2+\lambda\beta^T\beta\to\min_{\beta}$$

Stationarity condition can be written as:

$$2\sum_{n=1}^{N}x_n\left(x_n^T\beta-y_n\right)+2\lambda\beta=0$$

$$2X^T(X\beta-Y)+\lambda\beta=0$$

$$\left(X^TX+\lambda I\right)\beta=X^TY$$

so the solution is

$$\widehat{\beta}=(X^TX+\lambda I)^{-1}X^TY$$

Comments

- $X^T X + \lambda I$ is always non-degenerate as a sum of:
  - non-negative definite $X^T X$
  - positive definite λI
- Intuition:
  - out of all valid solutions select one giving simplest model
- Other regularizations also restrict the set of solutions.

Different account for different features

- Traditional approach regularizes all features uniformly:

$$\sum_{n=1}^{N}\left(x_n^T\beta - y_n\right)^2 + \lambda R(\beta) \to \min_w$$

- Suppose we have K groups of features with indices:

$$I_1, I_2, ...I_K$$

- We may control the impact of each group on the model by:

$$\sum_{n=1}^{N}\left(x_n^T\beta - y_n\right)^2 + \lambda_1 R(\{\beta_i | i \in I_1\}) + ... + \lambda_K R(\{\beta_i | i \in I_K\}) \to \min_w$$

- $\lambda_1, \lambda_2, ...\lambda_K$ can be set using cross-validation
- In practice: use standard regularizer but with different scaling of features.

Linear monotonic regression

- We can impose restrictions on coefficients such as non-negativity:

$$\begin{cases} L(\beta) = ||X\beta - Y||^2 \to \min_\beta \\ \beta_i \ge 0, \quad i = 1, 2, \ldots D \end{cases}$$

- Examples:
    - in credit scoring we know that salary should be positively correlated with credibility.
    - avaraging of forecasts of different prediction algorithms ($\beta_i = 0$ means, that i-th component does not improve accuracy of forecasting)

Table of Contents

Idea

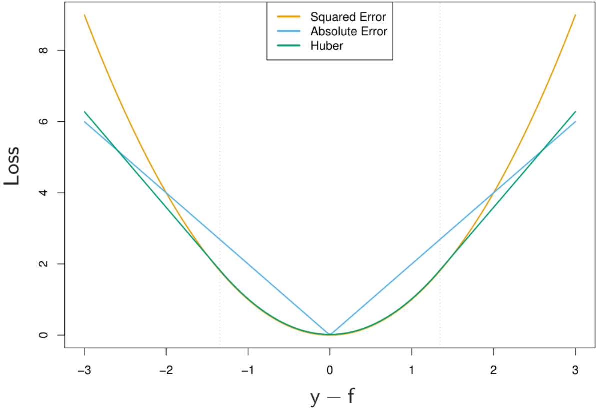
- Generalize quadratic to arbitrary loss:

$$\sum_{n=1}^{N}\left(x^T\beta - y_n\right)^2 \to \min_{\beta} \qquad \Longrightarrow \qquad \sum_{n=1}^{N}\mathcal{L}(x_n^T\beta - y_n) \to \min_{\beta}$$

LOSS	NAME	PROPERTIES
$\mathcal{L}(\varepsilon) = \varepsilon^2$	quadratic	differentiable
$\mathcal{L}(\varepsilon) = \lvert\varepsilon\rvert$	absolute	robust
$\mathcal{L}(\varepsilon) = \begin{cases}\frac{1}{2}\varepsilon^2, & \lvert\varepsilon\rvert \le \delta \\ \delta\left(\lvert\varepsilon\rvert - \frac{1}{2}\delta\right) & \lvert\varepsilon\rvert > \delta\end{cases}$	Huber	differentiable, robust

- Robust means solution is robust to outliers in the training set.

Non-quadratic loss functions



Optimal prediction for quadratic loss

Constant prediction $\widehat{y} \in \mathbb{R}$ for squared loss:

$$L(\widehat{y}) = \mathbb{E}\left\{(\widehat{y} - y)^2\right\} \to \min_{\widehat{y} \in \mathbb{R}}$$

$$\frac{\partial L(\widehat{y})}{\partial \widehat{y}} = \mathbb{E}\left\{2(\widehat{y} - y)\right\} = 2\widehat{y} - 2\mathbb{E}y = 0$$

$$\widehat{y} = \mathbb{E}y$$

Optimal prediction for absolute loss

Constant prediction $\widehat{y} \in \mathbb{R}$ for absolute loss:

$$L(\widehat{y}) = \mathbb{E}\{|\widehat{y} - y|\} = \int |\widehat{y} - y|\, p(y)dy =$$

$$= \int (\widehat{y} - y)\mathbb{I}[\widehat{y} \geq y]p(y)dy + \int (y - \widehat{y})\mathbb{I}[\widehat{y} < y]p(y)dy \to \min_{\widehat{y} \in \mathbb{R}}$$

$$\frac{\partial L(\widehat{y})}{\partial \widehat{y}} = \int \mathbb{I}[\widehat{y} \geq y]p(y)dy - \int \mathbb{I}[\widehat{y} < y]p(y)dy = 0$$

$$\frac{\partial L(\widehat{y})}{\partial \widehat{y}} = \int_{y \leq \widehat{y}} p(y)dx - \int_{y > \widehat{y}} p(y)dy = 0$$

$$\widehat{y} = \text{median}[y]$$

Loss function influences the result

- Consequently, for fixed x optimal prediction will be

$$\arg\min_{\widehat{y}(x)} \mathbb{E}\left\{ (\widehat{y}(x) - y)^2 \middle| x \right\} = \mathbb{E}[y|x]$$

$$\arg\min_{\widehat{y}(x)} \mathbb{E}\left\{ |\widehat{y}(x) - y| \,|\, x \right\} = \text{median}[y|x]$$

- For fixed training set and model result depends on the loss function.

Table of Contents

Weighted account for observations[2]

- Weighted account for observations

$$\sum_{n=1}^{N} w_n(x_n^T\beta - y_n)^2$$

- Weights may be used to:
    - decrease the impact of less reliable observations
        - e.g. outliers
    - make the unbalanced sample balanced
        - e.g. men and women in a hospital

[2] Derive solution for weighted regression.

Table of Contents

Support vector regression

Idea: don't care about small deviations, catch only the large ones + regularization.

$$\begin{cases} \frac{1}{2}\|w\|^2 \to \min_w \\ \langle w, x_n \rangle + w_0 - y_n \le \varepsilon \quad n = \overline{1, N} \\ y_n - \langle w, x_n \rangle - w_0 \le \varepsilon \quad n = \overline{1, N} \end{cases}$$

Since fitting any dataset with error$\in [-\varepsilon, \varepsilon]$ may be infeasible use penalization of excessive deviations:

$$\begin{cases} \frac{1}{2}\|w\|^2 + C\sum_{n=1}^{N}(\xi_n + \xi_n^*) \to \min_{w,\xi_n,\xi_n^*} \\ \langle w, x_n \rangle + w_0 - y_n \le \varepsilon + \xi_n, \quad \xi_n \ge 0 \qquad n = \overline{1, N} \\ y_n - \langle w, x_n \rangle - w_0 \le \varepsilon + \xi_n^*, \quad \xi_n^* \ge 0 \qquad n = \overline{1, N} \end{cases}$$

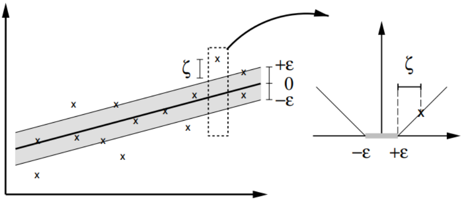
C controls how much errors should matter more than model simplicity.

Support vector regression

Equivalent unconstrained formulation:

$$\frac{1}{2}\|w\|^2 + C\sum_{n=1}^{N}\mathcal{L}(\langle w, x_n\rangle + w_0 - y_n) \to \min_w$$

with ε insensitive loss $\mathcal{L}(u) = \begin{cases} 0, & \text{if } |u| \le \varepsilon \\ |u| - \varepsilon & \text{otherwise} \end{cases}$



Solution will depend only on objects with $|\text{error}| \geq \varepsilon$, called *support vectors*.

Orthogonal matching pursuit

- Denote$\|w\|_0 = \#[\text{non-zero weights}]$
- Orhogonal matching pursuit finds approximate solution to

the problem:

$$\begin{cases} \|Xw - Y\|_2^2 \to \min_w \\ \|w\|_0 \le K \end{cases}$$

or equivalently (for $\varepsilon = \varepsilon(K)$)

$$\begin{cases} \|w\|_0 \to \min \\ \|Xw - Y\|_2^2 \le \varepsilon \end{cases}$$

Algorithm

1. Initialize model with constant zero, its residuals$=Y$
2. Repeat while $\|\beta\|_0 < K$ (or while $\|X\beta - Y\|_2^2 > \varepsilon$)
    1. add feature having maximum correlation with residuals
    2. fit multivariate regression: <u>selected features</u> vs. residuals
    3. update residuals by full account of features

- Method can be generalized
    - on any prediction algorithm
    - on any type of dependency measure between x and y

Summary

- Linear regression gives interpretable analytic solution.
- Non-linear dependencies can be modeled by adding non-linear features.
- Regularization:
    - allows working with linearly dependent features
    - smoothly controls model complexity
    - selects relevant features (Lasso, ElasticNet)
- Different loss functions yield different models and forecasts.