Regression Algorithms

- use the training data to estimate a fixed set of parameters, that will minimize some loss function
- Usually the loss function is a quadratic function (because if differenciable)
- The origin of the term "regression" to describe the process of fitting lines to data
- We can ajust to classification methods using a threshold (decision boundary), eighter hard or soft

Linear regression

- Parametric method
- We'll begin our study with univariate linear regression (1 variables), but then multivariate will be just an extension
- In linear regression, the space is convex
- Fitting a straight line:

$$h_w(x) = w_1 x + w_0 \tag{1}$$

- The task of finding the weights (parameters, or *coeficients*) that best fits these data
- weight space: the space defined by all possible settings of the weights
- Dependent variables: value to be predicted
- Independent variable: predictor value
- **Multivariate linear regression**
  - or *multiple regression*
  - extension of simple linear regression (now to more variables)
  - We'll have a data matrix

$$h_{sw}(\mathbf{x}_j) = w_0 + w_1 x_{j,1} + \cdots + w_n x_{j,n} = w_0 + \sum_i w_i x_{j,i} \, .$$

  - Variable selection
    - determining which predictors are associated with the response
    - Foward selection: begin the null model and increase the number of variables
    - Backward selection: start with all variables and them prune
  - Interaction
    - When one variable depend of another
    - We can model that including an extra term
    - If X1 depend of X2, we can model it as:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \epsilon$$

    - if we include an interaction in a model, we should also include the main effects (isolated variables), even if the p-values associated with their coefficients are not significant
    - To interection between at least one quantitative variable, the best aproach is to do an Regression Tree

- **least squares method**
  - to estimate the unknown linear regression coefficients
  - Computationally effycient
  - The regression will be a plane/hiperplane
  - Residual e (or *deviation*): difference of the predicted ($\hat{y}$) to the real value (y)
  - Sum of squared residuals:

$$\mathrm{RSS} = e_1^2 + e_2^2 + \cdots + e_n^2$$

$$e_i = y_i - \hat{y}_i \tag{2}$$

$$e_i = y_i - \beta_0 - \beta_1 x \tag{3}$$

  - The minimization of the RSS can be obtaining derivation, $\min_{\beta_0,\beta_1} \sum e_i^2$
  - That will result be:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2},$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x},$$

$\bar{y}$ and $\bar{x}$ are the mean of the training set

$$\hat{\beta}_1 = \frac{Cov(x,y)}{Var(x)} \tag{4}$$

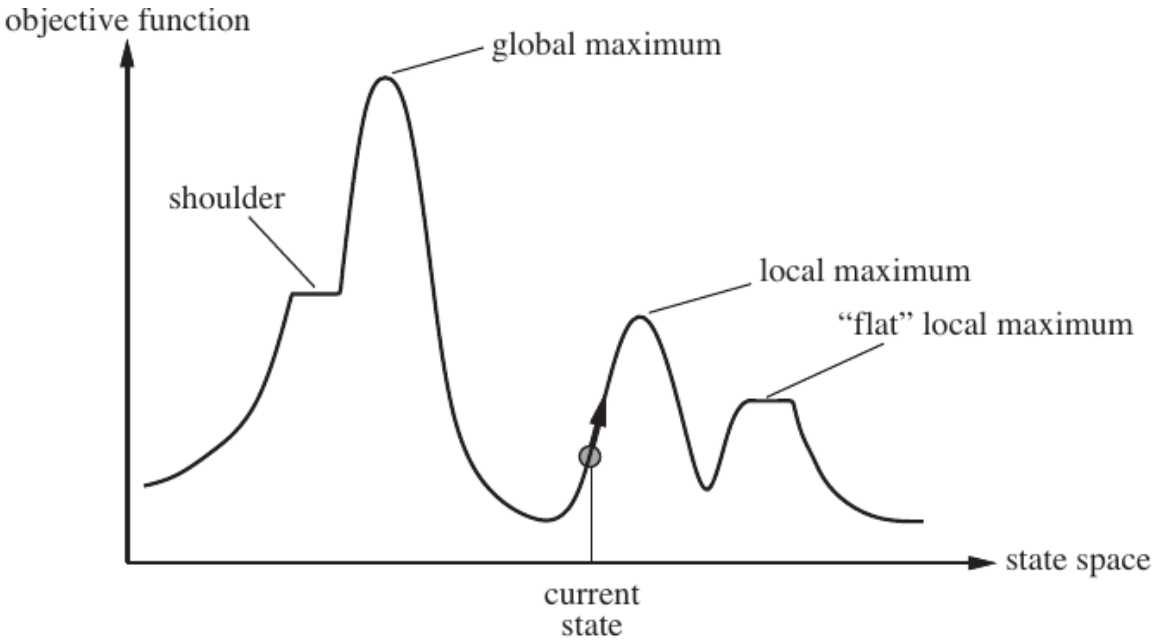
- **Gradient descent**
  - or *steepest descent*
  - pior do que o método dos quadrados mpinimos
  - O erro é calculado de forma quadrática para penalizar erros distantes, além de manter a diferenciabilidade, além de que se simplemente tirássemos o módulos dos dados daria ruim
  - Se encontrarmos um conto de parametros que minimizam o erro para o conjunto de dados, provavelmetne seremos capazes de extrapolar isso
  - The theory is almost equal to Hill Climbing (but here we'll deal just with differenciable functions)
  - minimize some function by iteratively moving in the direction of steepest descent (the negative of the gradient)
  - algorithm:

$\mathbf{w} \leftarrow$ any point in the parameter space
loop until convergence **do**
  for each w_i **in w do**

$$w_i \leftarrow w_i - \alpha \frac{\partial}{\partial w_i} Loss(\mathbf{w})$$

  - *Practical Tip:* the algorithm genarally converges faster with normalized features

- Learning rate (α)
  - Size of each etep
  - high learning rate cover more “ground” each step, but we risk overshooting the lowest point
  - A low learning rate is more precise, but slower
  - Can be constant or decay over time
- Variation: batch gradient descent
  - Variation in wicth unique global minimum is guaranteed, but is slow
- Variation: stochastic gradient descent
  - Does not guaranteed, but is faster
- Convergence
  - If the space is convex, it converges to the global solution

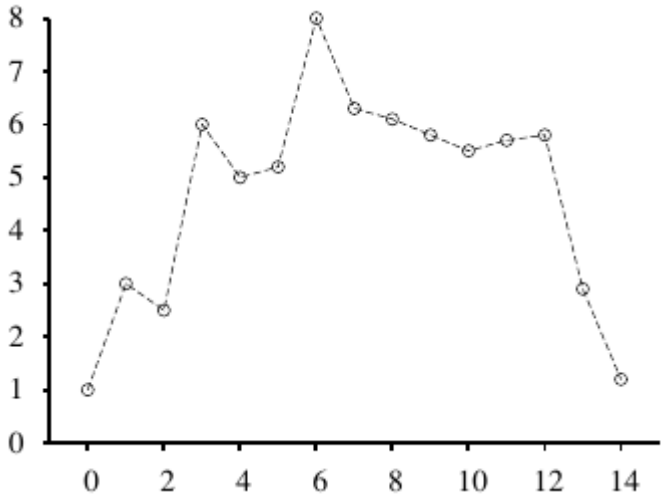


- **Commom problems**
  - Non-constant Variance of Error
    - error increase with (per example) increasing X_1
    - A good way to detect them is through an residual plot (residual vs X_1)
    - If the error increase with X1, a good aproach is linearize the input doing $X_2 = log(X_1)$ or $X_2 = \sqrt{X_1}$
  - High leverage point
    - Leverage = *alavancagem*
    - An value that is outside of the normal range of the observations will have high impact on the fit
    - Noise in that point can cause real problems
  - Collinearity
    - two or more predictor variablescollinearity are closely related to one another
    - A simple way to detect collinearity is to look at the correlation matrix
    - not all collinearity problems can be detected by inspection of the correlation matrix (ex: beetween two or more, or with some threshold)

    - Solutions: Drop some variables, unify them or include that correlation in the model (described in *Multivariable regression - interaction*)

Others regressions

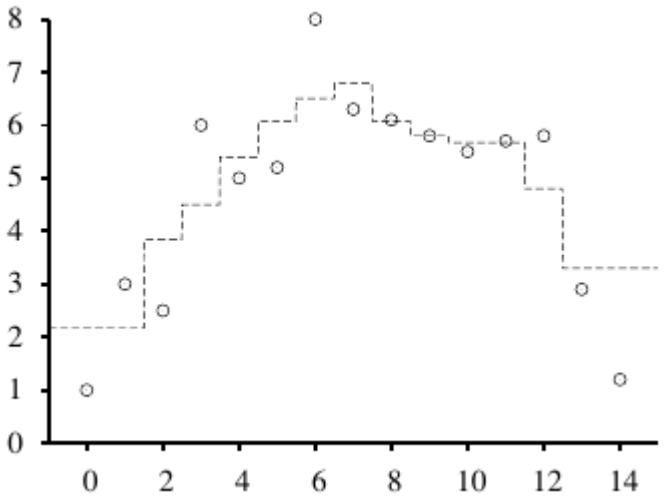
- **Polinomial regression**
    - Add a new variables, elevating an old variable (ex: $X_2 = X_1^3$)
    - The math will be the same as a linear regression
- **piecewise-linear regression**
    - Non Parametric method
    - “connect-the-dots”
    - when given an input, solves the ordinary linear regression problem with two points (the nearest to let and right)
    - Usage: low noise data
    - (-) discontinuos
    - *Example:*



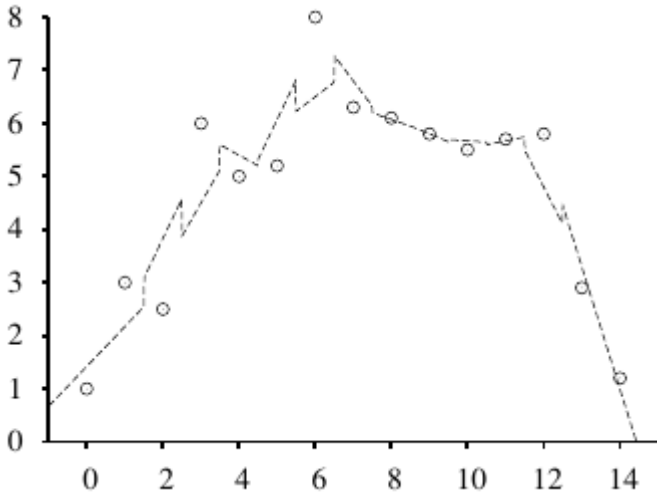
- **nearest neighbors average**
    - Non Parametric method
    - h(x) is the mean value of the k points
    - (-) the estimates are poor in fast variation points
    - (-) discontinuos

$$\hat{f}(x_0) = \frac{1}{K} \sum_{x_i \in \mathcal{N}_0} y_i$$

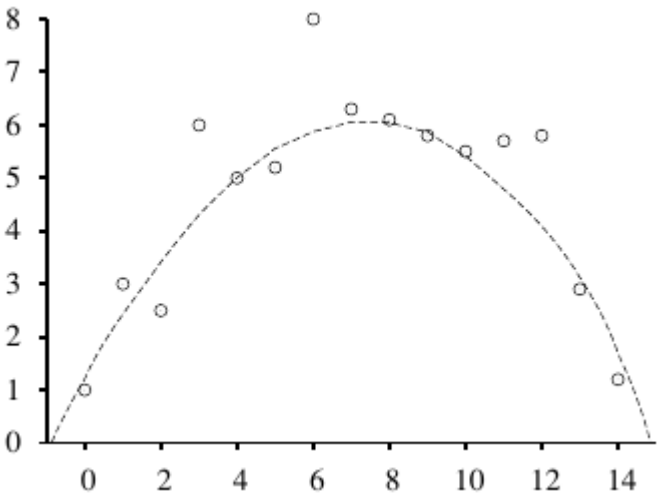
$\mathbb{N}_0$ is the set of k nearest neighbors



- **nearest-neighbors linear regression**
  - Non Parametric method
  - improves on connect-the-dots
  - (-) discontinuos
  - *Example:*



- **locally weighted regression**
  - Non Parametric method
  - The idea of locally weighted regression is that at each query point xq , the examples that are close to xq are weighted heavily, and the examples that are farther away are weighted less heavily or not at all
  - *Example:*



    - Kernel function: decide how much to weight each example
- **Poisson regression method**
    - models integer count data
- **Spline**
    - ?
- **Arvore de regressão**
    - ou *Arvore de modelos*
    - To more information, see *decision tree*
    - As folhas são equações de regressão
    - A regression tree has at each leaf a linear function of some subset of numerical attributes, rather than a single value

Avaliação de modelos de regressão

- **Mean Absolute Error** (MAE)
    - it's the average error
    - $\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$
- **Mean Squared Error** (MSE)
    - "punishes" larger errors
    - $\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
- **Root Mean Squared Error** (RMSE)
    - is interpretable in the "y" units
    - $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$
- **Residual standard error (RSE)**
    - métrica de avaliação para regressão linear
    - measure of the lack of fit of the model to the data
    - standar deviation of e
    - Better if equal to zero
- **Coefficient of determination (R^2)**
    - métrica de avaliação para regressão linear
    - is the fraction of variability explained by the model

    - Quão bem o modelo (como um todo) explica os valores da variavel independente
    - ExplainedVariation / TotalVariation
    - When using least square methods, R=p (pearson correlation)
    - Obtido a partir da diferença entre um conjunto de valores reiais y e um conjunto de valores previstos f
    - Se o modelo for realmente linear, o erro provavelmente terá uma distribuição gaussiana com média zero
    - quanto mais proximo de 1, melhor

$$R^2 = 1 - \frac{Var(e)}{Var(y)} \tag{5}$$

- Loss functions based on mean: minimiza o erro quadrático
- Loss functions based on median: minimiza o desvio absoluto