Aprententatge Automàtic per a Xarxes (ML4Net)

Seminar 1 - Linear regression

Francesc Wilhelmi & Boris Bellalta
(Credits to Marc Carrascosa)

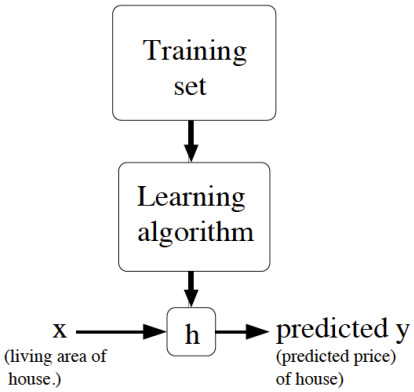
School of Engineering
Universitat Pompeu Fabra

Supervised learning

Main idea: Learn a function that maps features (inputs) to labels (outputs).

Key concepts:

- Features (x): Input information for the predictions.
- Labels (y): Variables to be predicted.
- Model (h): Function $h(x) = y$ (set of weights and parameters) that maps x to y.
- Train/test data: Partitions made to train and validate a model.
- Cost function (J): Measures the difference between the model's predictions and the actual labels (e.g., Mean Average Error (MAE)).
- Prediction error/accuracy: Difference between the model predictions and the actual values (test data).
- Optimizer: Criteria that we follow to update the model weights towards minimizing the cost function.

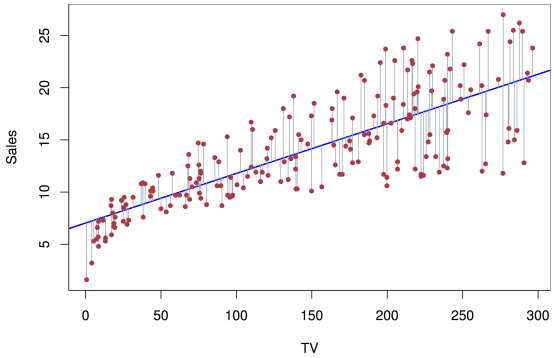


Logistic regression

Hypothesis:

$$y = h_\theta(x) = \theta^T x = \theta_0 + \theta_1 x_1 + ... + \theta_n x_n,$$

where y is the target value to be predicted, x is the features vector, and θ is the parameters (weights) to be learned.



Check `https://kwichmann.github.io/ml_sandbox/linear_regression_diagnostics/`

Learning objective

- The goal is to learn a function $h(x) = \hat{y}$ that approximates $\hat{y} \approx y$.
- For that, we use a given dataset as a reference (learn from experience).
- This means that we learn a set of weights (θ) that minimize a cost function (we fit a linear equation to some data).
- In our case, we focus on the 'least squares' cost function:

$$J(\theta) = \frac{1}{2} \sum_{i=0}^{M} (h(x_i) - y_i)^2$$

- To minimize the cost function, we build an 'optimizer' (here, Gradient descent), which will try different values of θ according to some criteria.

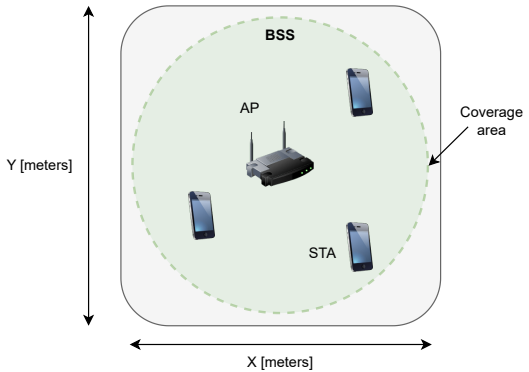
Dataset (I)

How were the data generated?

- Simulated Wi-Fi deployments using random values for certain parameters
- Different Wi-Fi features are considered: number of STAs, traffic load, CW, etc.
- Performance metrics (e.g., throughput, delay) are outputted to be used as labels (i.e., prediction goal).

How are the data structured?

- Columns are features
- Rows are samples



Dataset (II)

Features:

- Number of STAs: 1-40
- Load: 0.5 - 82 Mbps
- Size(x): 1-40 m
- Size(y): 1-40 m
- Area: xy
- Contention window: {3, 15, 31, 63, 127, 255, 511, 1023} slots
- Channel width: {20, 40, 80, 160} MHz
- Packet size: {4000, 6000, 8000, 10000, 12000} bits
- Max RSSI: $\geq$ -82 dBm
- Avg. RSSI: $\geq$ -82 dBm
- Min. RSSI: $\geq$ -82 dBm

Labels:

- Avg. Probability of failure: 0-1
- Throughput: Aggregated over the whole network, in bps
- Average delay: Average transmission time of a packet, in seconds
- Total airtime: Sum of transmission time necessary for all transmissions
- Proportional airtime: Available transmission time used by the network

Focus on the **throughput**