Fondements de l'apprentissage machine

Automne 2014

Roland Memisevic

Leçon 2

Plan

- Régression linéaire
- Apprentissage par l'optimisation
- Apprentissage par maximum vraisemblance
- La décomposition biais–variance
- Un premier aperçu de la modelisation Bayesiénne

Régression linéaire

- La régression linéaire est un des concepts les plus fondamentaux de l'apprentissage machine et statistiques.
- Il s'agit d'un simple problème avec une solution simple.
- Pourtant, elle nous permet d'etudier une variété de concepts aux centre de l'apprentissage machine, avec lequelles nous serons occupés pendant ce cours.
- En raison de sa simplicité, la régression linéaire est également utilisé dans très nombreuses tâches pratiques.

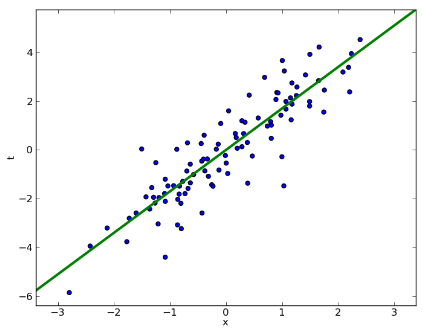
Régression linéaire

$$\mathbf{x} \rightarrow \mathbf{t}$$

- Nous sommes donnés un ensemble d'observations $\mathbf{x}$ et $\mathbf{t}$ à valeurs réelles.
- Probléme: *Apprendre à prédire* $\mathbf{t}$ *de* $\mathbf{x}$.
- Il s'agit d'un problème *d'apprentissage supervisé*.

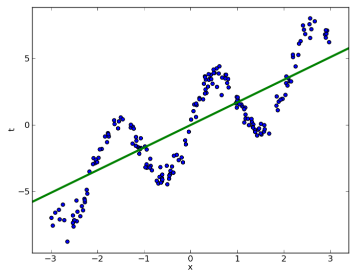
Régression linéaire 1-d

- Si les entrées et les sorties sont des scalaires (1-d), on peut les visualiser:



- La régression linéaire est basée sur l'hypothèse qu'il existe une relation linéaire entre $\mathbf{x}$ et $\mathbf{t}$.

Bruit vs. dependances dont on ne se soucie pas



- Même si cette hypothèse n'est pas tout á fait correct, on peut utiliser la régression linéaire, et capturer la *tendance linéaire* dans les données (en déclarant toutes les dépendances non-linéaires comme le bruit)

Regression en 1-d

- Pour des entrées/sorties 1-d on a:

$$y(x) = w_0 + w_1 x$$

- Pour des entrées en D dimensions et des sorties 1-d on a:

$$y(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + \ldots + w_D x_D \big(= w_0 + \mathbf{w}^{\mathrm{T}} \mathbf{x} \big)$$

- Pour des sorties en K dimensions on a simplement un modéle pour chaque dimension de $\boldsymbol{y}$:

$$\begin{pmatrix} y_1(\mathbf{x}) \\ \vdots \\ y_K(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} y_1(\mathbf{x}) = w_{10} + w_{11} x_1 + w_{12} x_2 + \ldots + w_{1D} x_D \\ \vdots \\ y_K(\mathbf{x}) = w_{K0} + w_{K1} x_1 + w_{K2} x_2 + \ldots + w_{KD} x_D \end{pmatrix}$$

Le terme de biais

- w_0 s'appelle "biais" ("bias" en anglais). Il nous permet de deplacer le modéle linéaire a travers l'axe des $\boldsymbol{y}$.
- On peut toujours éliminer le "biais" (et c'est courant) en remplacant:

$$\begin{pmatrix} x_1 \\ \vdots \\ x_D \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ x_1 \\ \vdots \\ x_D \end{pmatrix}$$

Maintenant le modèle linéaire contient le terme de biais implicitement (ce qui nous permet d'écrire $\mathbf{w}^{\mathrm{T}}\mathbf{x}$).

Méthode des moindres carrés

- Estimation des paramètres ("poids"). (1-d sorties pour le moment)
- Pour estimer les paramètres il faut avoir un ensemble d'entrainement

$$\mathcal{D} = \{(\mathbf{x}_n, t_n)\}_{n=1}^{N}$$

dont on minimise le *somme-des-erreurs carrés*:

$$E(\mathbf{w}; \mathcal{D}) = \frac{1}{2}\sum_{n=1}^{N}\left(t_n - \mathbf{w}^{\mathrm{T}}\mathbf{x}_n\right)^2$$

par rapport à $\mathbf{w}$.

- Des autres function des pertes sont possible. Mais celui-ci rend l'optimisation le plus facile.

Méthode des moindres carrés

- Pour l'optimiser par rapport à $\mathbf{w}$ on difference

$$\frac{\partial E}{\partial \mathbf{w}} = -\sum_{n=1}^{N}(t_n - \mathbf{w}^{\mathrm{T}}\mathbf{x}_n)\mathbf{x}_n^{\mathrm{T}}$$

- Définissant le dérivé a zéro

$$0 = -\sum_{n=1}^{N} t_n \mathbf{x}_n^{\mathrm{T}} + \mathbf{w}^{\mathrm{T}}\Big(\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^{\mathrm{T}}\Big)$$

on obtient:

$$\mathbf{w} = \Big(\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^{\mathrm{T}}\Big)^{-1}\Big(\sum_{n=1}^{N} t_n\mathbf{x}_n^{\mathrm{T}}\Big)$$

- (Il est plus simple et éducatif de le faire pour les entrées 1-d.)

Méthode des moindres carrés

- Pour le formuler plus compactement definir $\mathbf{t}$ le vecteur de sorties

$$\mathbf{t} = \begin{pmatrix} t_1 \\ \vdots \\ t_N \end{pmatrix}$$

ainsi que $\mathbf{X}$ une matrice de entrées (une par rangée):

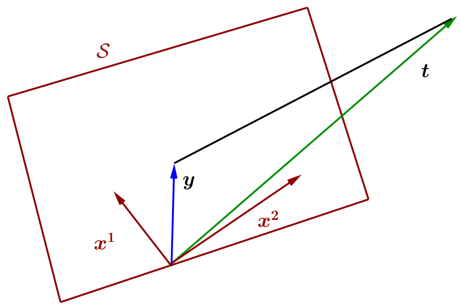
$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_N \end{pmatrix}$$

Cela nous permet d'écrire la solution plus compacte:

Les équations normales

$$\mathbf{w} = \left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{t}$$

Interprétation géométrique



- On peut interpéter l'erreur comme la norme au carré de la différence entre vecteurs $\mathbf{t}$ et $\mathbf{y}$, contenant tous les sorties et tous les prédictions, respectivement.
- Le vecteur $\mathbf{y}$ $\left(= \mathbf{X}\mathbf{w}\right)$ réside dans le sous-espace $\mathcal{S}$ engendré par les colonnes $\mathbf{x}^i$ de $\mathbf{X}$.
- Il s'agit de la *projection orthogonale* de $\mathbf{t}$ sur $\mathcal{S}$.