CSC2515: Midterm Review

Ekansh Sharma

October 23, 2019

Built from slides by James Lucas

Midterm Test

- Time: Wednesday, Oct. 30, from 4:10-5:40pm
- Location: Health Sciences building, room 610
- Office Hours: Fri 10/25, 12-1pm,6-7pm in BA3201
  Mon 10/28, 11am-noon, in BA3201
  Tue 10/29, 2-4pm, in BA3201
  Wed 10/30, noon-1pm, in BA1190

Agenda

1. A brief overview
2. Some sample questions

Basic ML Terminology

- Regression
- Overfitting
- Generalization
- Bias–Variance
- Bayes Optimal
- Classification
- Underfitting
- Regularization
- Bayes Error
- Stochastic Gradient Descent (SGD)

Basic ML Terminology

- Model
- Linear classifier
- Training Data
- Optimization
- 0-1 Loss
- Validation Data
- Convexity
- Features
- Test Data

Some Questions

Question 1

Given $\{(x_i, t_i)\} = S \sim \mathcal{D}$. Let h_S be the predictor for dataset S. Given x,

1. Bias of the predictor is $(\mathbb{E}_S h_S(x) - \mathbb{E}[t|x])^2$
2. Bayes error is $\mathrm{Var}[t|x]$

Question2

Take labelled data $(\mathbf{X}, \mathbf{y})$.

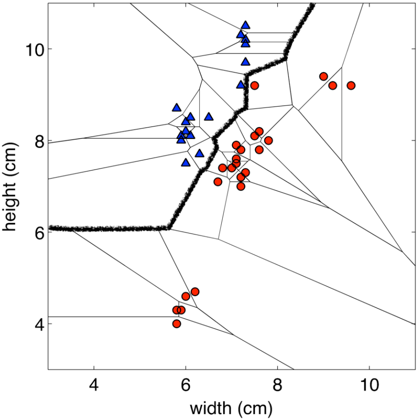
1. Why should you use a validation set?
2. How do you know if your model is overfitting?
3. How do you know if your model is underfitting?

Topics covered so far...

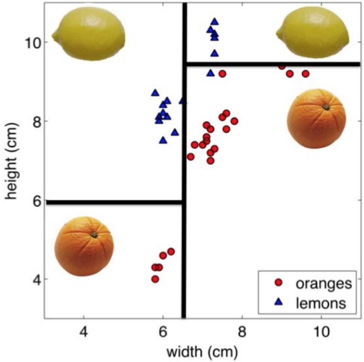
1. Nearest Neighbours
2. Decision Trees
3. Ensembles
4. Linear Regression
5. Linear Classification
6. SVMs
7. Neural Networks

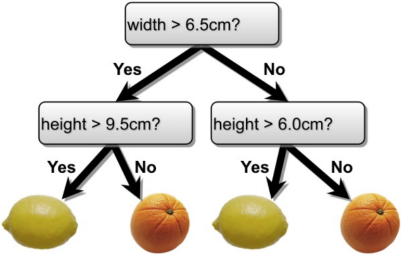
Nearest Neighbours

1. Decision Boundaries
2. Choice of 'k' vs. Generalization
3. Curse of dimensionality



Decision Trees





1. Entropy/Information Gain
2. Decision Boundaries

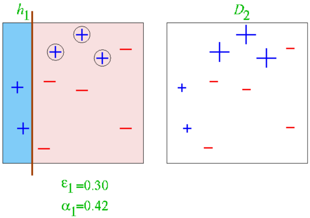
Ensemble Methods

Bagging

1. Bias-Variance tradeoff
2. Average the predictions of *m* models trained on bootstrapped datasets.
3. Random Forest

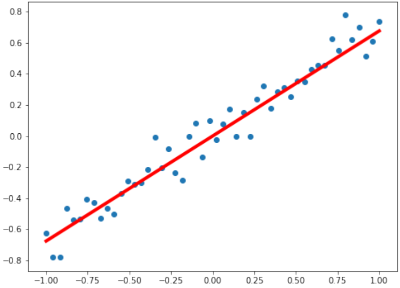
Boosting

1. Sequentially train *weak classifiers*.
2. Additive model with exponential loss



Linear Regression

1. Model: $\mathbf{y} = \mathbf{w}\mathbf{x} + \mathbf{b}$
2. Objective: Minimize squared loss
3. Direct Solution
4. (Stochastic) Gradient Descent
5. Regularization



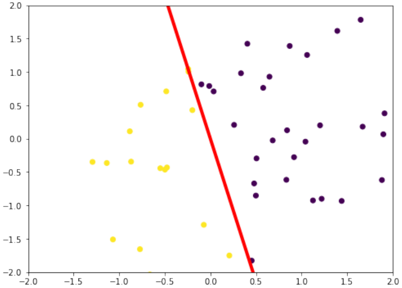
Linear Classification

Binary Linear Classification

1. Model: $z = \mathbf{w}\mathbf{x} + b$,
   $y = \mathbb{I}(z \geq 0)$
2. Objective: Minimize 0-1 loss
3. Surrogate loss

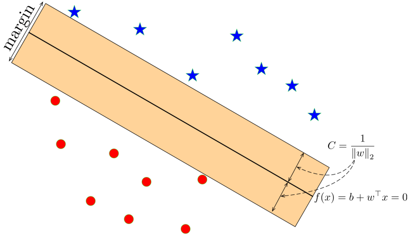
Logistic Regression

1. Model: $z = \mathbf{w}\mathbf{x} + b$,
   $y = \sigma(z)$
2. Objective: Minimize cross-entropy
3. Multi-class classification with softmax function



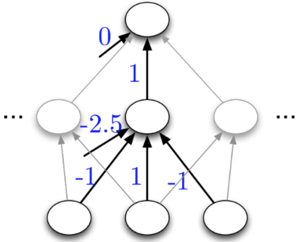
Linear Classification: SVMs

1. Model: $y = \operatorname{sign}(\mathbf{wx} + b)$
2. Objective: Maximize *margin*.
3. Soft-margin SVM: Linear classifier with hinge loss and ℓ_2-regularization.



Neural Networks

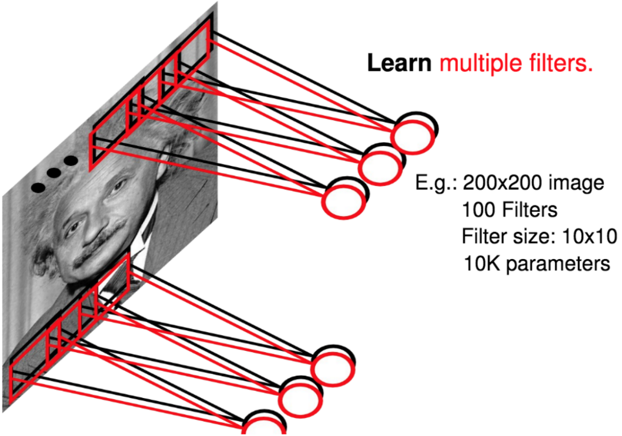
1. Model: $y = f^{(L)} \circ \cdots \circ f^{(1)}(x)$
2. Weights and activation functions
3. Depth and expressive power
4. Backpropagation
5. Non-convex problem



from slides by James Lucas and David Madras

Convolutional Neural Networks

1. Convolutional Neural Networks (CNN) Architecture
2. Local connections/convolutions/pooling
3. Feature Learning



from slides by James Lucas and David Madras

Sample Question 1

Assume we are preprocessing our data using an **invertible** linear transformation on the features of our training data. The transformation can either be some orthogonal (i.e. rotations) matrix or some diagonal matrix.

Say if this can have any effect on the performance of the following algorithms, and explain in no more than two sentences.

- Orthogonal preprocessing on decision tree classification.
- Diagonal preprocessing on decision tree classification.
- Orthogonal preprocessing on nearest neighbor classification.
- Diagonal preprocessing on nearest neighbor classification.

Q1 Solution

- Orthogonal preprocessing on decision tree classification.
  Will have an effect. Rotation changes the axis.
- Diagonal preprocessing on decision tree classification.
  Will not have an effect. Rescaling along axis will shift split criteria but wont change decision.
- Orthogonal preprocessing on nearest neighbor classification.
  Will not have an effect. Orthogonal linear transformations will preserve distances.
- Diagonal preprocessing on nearest neighbor classification.
  Will have an effect. Will change distances between data points.

Sample Question 2

Given input $\mathbf{x} \in \mathbb{R}^d$ and target $y \in \mathbb{R}$, define $\hat{\mathbf{x}} = \mathbf{x} + \boldsymbol{\epsilon}$ to be a noisy pertubation of $\mathbf{x}$ where we assume

- $\mathbb{E}[\epsilon_i] = 0$
- for $i \neq j$: $\mathbb{E}[\epsilon_i \epsilon_j] = 0$
- $\mathbb{E}[\epsilon_i^2] = \lambda$

We define the following objective that tries to be robust to noise

$$\mathbf{w}^* = \arg\min \mathbb{E}_\epsilon[(\mathbf{w}^T \hat{\mathbf{x}} - y)^2] \tag{1}$$

Show that it is equivalent to minimizing L_2 regularized linear regression, i.e.

$$\mathbf{w}^* = \arg\min \left[(\mathbf{w}^T \mathbf{x} - y)^2 + \lambda ||\mathbf{w}||^2\right] \tag{2}$$

Q2 Solution

We can write the inner term as,

$$\begin{aligned}(\mathbf{w}^T\hat{\mathbf{x}} - y)^2 &= (\mathbf{w}^T\mathbf{x} + \mathbf{w}^T\boldsymbol{\epsilon} - y)^2 && (3)\\ &= (\mathbf{w}^T\mathbf{x} - y)^2 + 2\mathbf{w}^T\boldsymbol{\epsilon}(\mathbf{w}^T\mathbf{x} - y) + (\mathbf{w}^T\boldsymbol{\epsilon})^2 && (4)\\ &= (\mathbf{w}^T\mathbf{x} - y)^2 + 2\mathbf{w}^T\boldsymbol{\epsilon}(\mathbf{w}^T\mathbf{x} - y) + (\mathbf{w}^T\boldsymbol{\epsilon}^T\boldsymbol{\epsilon}\mathbf{w}) && (5)\end{aligned}$$

Under the expectation the second term will be zero as it is a linear combination of the elements of $\boldsymbol{\epsilon}$. The final term will be the quadratic form of $\mathbf{w}$ with the covariance of $\boldsymbol{\epsilon}$. The covariance is simply λI. Thus we are minimizing,

$$(\mathbf{w}^T\mathbf{x} - y)^2 + \lambda||\mathbf{w}||^2$$

which is exactly the objective of L2-regularized linear regression.