1. Text Classification

Supervised learning basics

Empirical risk minimization (ERM)

We want to build a model: h : $\mathcal{X}$ (input space) → $\mathcal{Y}$ (output space)

- Assume a data generating distribution D over $\mathcal{X} \times \mathcal{Y}$
- We have access to a training set: m samples from $D\{(x^{(i)}, y^{(i)}\}_{i=1}^{m}$
- We can measure the goodness of a prediction $h(x)$ by comparing it against the ground truth $y$using some **loss function**
- Our goal is to minimize the expected loss over D **(risk)**:

  minimize $\mathbb{E}_{(x,y)\sim D}\left[\text{error}(h, x, y)\right]$

  but it **cannot be computed**
- Instead, we minimize the average loss on the training set **(empirical risk)**:

  minimize $\frac{1}{m}\sum_{i=1}^{m}\text{error}(h, x^{(i)}, y^{(i)})$

Overfitting vs underfitting

- Trivial solution to (unconstrained) ERM: memorize the data points
- Solution: constrain the prediction function to a subset, i.e. a hypothesis space $h \in H$

Summary

1. Obtain training data $D_{\text{train}} = \{(x^{(i)}), y^{(i)})\}_{i=1}^{n}$
2. Choose a loss function L and a hypothesis class H
3. Learn a predictor by minimizing the empirical risk

Generative models: naive Bayes

Text classification

- Input: text (sentence, paragraph, document)
- Predict the category or property of the input text

Problem formulation

- Input: a sequence of tokens $x = (x_1, ...x_n)$ where $x_i \in \nu$.
- Output: binary label $y \in \{0, 1\}$.
- Probabilistic model:

$$f(x) = \begin{cases} 1 & p_\theta(y = 1|x) > 0.5 \\ 0 & \text{otherwise} \end{cases}$$

where p_θ is a distribution parametrized by $\theta \in \Theta$.

Naive Bayes assumption: The input features are **conditionally independent** given the lable:$p(x|y) = \prod_{i=1}^{n} p(x_i|y)$

- A strong assumption, but works surprisingly well in practice

Learning: maximum likelihood estimation

Likelihood function of θ given D:

$L(\theta; D) \stackrel{\text{def}}{=} p(D; \theta) = \prod_{i=1}^{n} p(y_i; \theta)$

Maximum (log-)likelihood estimator:

$\hat{\theta} = \arg\max_{\theta \in \Theta} L(\theta; D) = \arg\max_{\theta \in \Theta} \sum_{i=1}^{n} \log p(y_i; \theta)$

ERM: $\min \sum_{i=1}^{N} l(x^{(i)}, y^{(i)}, \theta)$

MLE: $\max \sum_{i=1}^{N} \log p(y^{(i)}|x^{(i)}; \theta)$

MLE is equivalent to ERM with the **negative log-likelihood** (NLL) loss function: $l_{\text{NLL}}(x^{(i)}, y^{(i)}, \theta) \stackrel{\text{def}}{=} -\log p(y^{(i)}|x^{(i)}; \theta)$

Inference: make predictions using the model

$y = \arg\max_{y \in Y} p_\theta(y|x)$

Discriminative models: logistic regression

	generative models	discriminaive models
modeling	joint: $p(x, y)$	conditional: $p(y|x)$
assumption on y	yes	yes
assumption on x	yes	no
development	generative story	feature extractor

Map $w \cdot \phi(x) \in \mathbb{R}$ to a probability by the logistic function

Binary: $p(y = 1|x; w) = \frac{1}{1+e^{-w \cdot \phi(x)}}$ $(y \in \{0, 1\})$

Multiclass: $p(y = k|x; w) = \frac{e^{w_k \cdot \phi(x)}}{\sum_{i \in \mathcal{Y}} e^{w_i \cdot \phi(x)}}$ $(y \in \{1, \dots, K\})$ “softmax”

Inference:

$\hat{y} = \arg\max_{k \in \mathcal{Y}} p_\theta(y = k|x; w) = \arg\max_{k \in \mathcal{Y}} w_k \cdot \phi(x)$

BoW representation: a sentence is the “sum” of words

N-gram features: continuous sequences of n words

Regularization, model selection, evaluation

Error decomposition

$\text{risk}(\hat{h}) - \text{risk}(h^*) = \text{approximation error} + \text{estimation error}$

- Approximation error: $\text{risk}(\text{best hypo in } H) - \text{risk}(h^*)$

  Does my hypothesis space contain the true hypothesis?

- Estimation error: $\text{risk}(\hat{h}) - \text{risk}(\text{best hypo in } H)$

  Can I find the best hypothesis given limited data?

Larger hypothesis class: approximation error ↓, estimation error ↑

Smaller hypothesis class: approximation error ↑, estimation error ↓

Reduce the dimensionality

Linear predictors: reduce the number of features $H = \{w : w \in \mathbb{R}^d\}$

For other predictors: depth of decision trees, degree of polynomials, number of decision stumps in boosting…

Regularization

Regularization: reduce the “size” of w

$\min \frac{1}{N} \sum_{i=1}^{N} l(x^{(i)}, y^{(i)}, w) + \frac{\lambda}{2} ||w||_2^2$

Validation

Validation set: a subset of the training data reserved for tuning the learning algorithm

K-fold cross validation