machine learning hypothesis evaluation

evaluation a learning algorithm

after implementing regularized linear regression to predict a target

$$J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 + \lambda\sum_{j=1}^{m}\theta_j^2\right]$$

large prediction errors can be addressed with the following methods:
obtain additional training examples
more carefully selected smaller set of features $(x_1, x_2, x_3, \dots, x_{100})$
obtaining additional features
adding polynomial features (x_1^2, x_1^2, x_1x_2, etc.)
decreasing λ
increasing λ

machine learning diagnostics

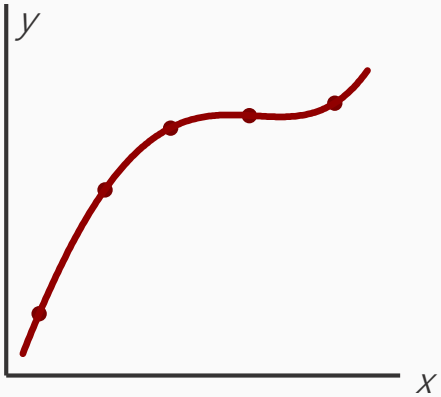
diagnostic: a test run to gain insight on what is and is not working with a learning algorithm; gaining guidance on how to best improve its performance.

machine learning diagnostics can be time consuming to implement, however, doing so can save significant time in design as opposed to applying the above examples blindly

machine learning diagnostics can rule out certain courses of action to change the learning algorithm when evaluated as being unlikely to significantly improve performance

evaluating a hypothesis

evaluating a hypothesis



$$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

low training error fails to generalize to new examples outside of the training set · offers poor predictive power

randomly partitioning a dataset into a training and test set (~70%/30% split) allows a model to be evaluated for accuracy (ability to generalize)

$$\begin{matrix}\left(x^{(1)}, y^{(1)}\right)\\ \vdots\\ \left(x^{(m)}, y^{(m)}\right)\end{matrix}$$
$$\begin{matrix}\left(x_{test}^{(1)}, y_{test}^{(1)}\right)\\ \vdots\\ \left(x_{test}^{(m)}, y_{test}^{(m)}\right)\end{matrix}$$

example
implementation of linear regression (without regularization) will overfit the training set if the training error $J(\theta)$ is **low** and the test error $J_{test}(\theta)$ is **high**

training/testing procedure for linear regression

learn parameter θ from training data (minimizing training error $J(\theta)$)
compute test error using the squared error metric:

$$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} \left(h_\theta \left(x_{test}^{(i)} \right) - y_{test}^{(i)} \right)^2$$

training/testing procedure for logistic regression

learn parameter θ from training data (minimizing training error $J(\theta)$)
compute test error using the squared error metric:

$$J_{test}(\theta) = -\frac{1}{m_{test}} \left[\sum_{i=1}^{m_{test}} y_{test}^{(i)} \log h_\theta \left(x_{test}^{(i)} \right) + \left(1 - y_{test}^{(i)} \right) \log \left(1 - h_\theta \left(x_{test}^{(i)} \right) \right) \right]$$

alternative test error metric: misclassification error (0/1)

$$\text{err}(h_\theta(x), y) = \begin{cases} 1 & \text{if } h_\theta(x) \geq 0.5, \quad y = 0 \\ & \text{or if } h_\theta(x) \leq 0.5, \quad y = 1 \\ 0 & \text{otherwise} \end{cases}$$

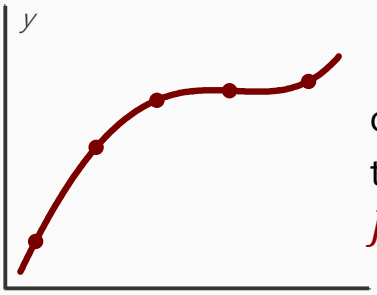
$$\text{test error} = \frac{1}{m_{test}} \sum_{i=1}^{m_{test}} \text{err} \left(h_\theta \left(x_{test}^{(i)} \right), y_{test}^{(i)} \right)$$

the misclassification error during training is often a standard approach to evaluating the accuracy of a model. a model with excessive false positives/negatives will yield low predictive accuracy. this method is commonly displayed in a classification matrix

$$\begin{bmatrix} \text{true positive} & \text{false positive} \\ \text{true negative} & \text{false negative} \end{bmatrix} \text{ or } \begin{bmatrix} \text{true positive} & \text{false positive} \\ \text{false negative} & \text{true negative} \end{bmatrix} \text{ etc.}$$

model selection and train/validation/test sets

determining what degree of polynomial to fit a dataset, what features to include in an algorithm, and what regularization parameter for a learning algorithm are components to the model selection process. Partitioning data into training, validation, and test sets.



$$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

once parameters $\theta_0, \theta_1, \dots, \theta_4$ fit to some set of data (training set), the error of parameters measured on that data (the training error $J(\theta)$) is likely to be lower than the actual generalization error

model selection

determining what degree of polynomial (linear, quadratic, cubic, etc.) to fit to the data

a hypothetical additional parameter to (*d*) to represent the best degree of polynomial

first option is to minimize the training error of each model for parameter vector $\theta^{(xi)}$

from the parameter vectors, test set error can be calculated, choosing the lowest error

$$\begin{aligned}
d = 1 &\rightarrow h_\theta(x) = \theta_0 + \theta_1 x &&\rightarrow \theta^{(1)} &&\rightarrow J_{test}(\theta^{(1)}) \\
d = 2 &\rightarrow h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 &&\rightarrow \theta^{(2)} &&\rightarrow J_{test}(\theta^{(2)}) \\
d = 3 &\rightarrow h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3 &&\rightarrow \theta^{(3)} &&\rightarrow J_{test}(\theta^{(3)}) \\
&\vdots \quad\quad\quad\quad \vdots \\
d = 10 &\rightarrow h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10} &&\rightarrow \theta^{(10)} &&\rightarrow J_{test}(\theta^{(10)})
\end{aligned}$$

choose $\theta_0 + \theta_1 x + \cdots + \theta_5 x^5$

how well does the model generalize: report test set error $J_{test}(\theta^{(5)})$

problem: $J_{test}(\theta^{(5)})$ is a likely optimistic estimate of generalization error

(i.e. the parameter (*d* = degree of polynomial is fit to the **test set**)

parameters developed to determine predictive power on the test set is less likely to fit the data to samples never seen before by the model. overfitting the test set is common

in addition to a training and test set, an additional validation set can prevent overfitting

possibly a ~60%/20%/20% split as opposed to the 70%/30% split previously referred

train · validation · test error

training error:

$$J_{train}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right), y^{(i)}\right)^2$$

cross validation error:

$$J_{cv}(\theta) = \frac{1}{2m_{cv}}\sum_{i=1}^{m_{cv}}\left(h_\theta\left(x_{cv}^{(i)}\right), y_{cv}^{(i)}\right)^2$$

test error:

$$J_{test}(\theta) = \frac{1}{2m_{test}}\sum_{i=1}^{m_{test}}\left(h_\theta\left(x_{test}^{(i)}\right), y_{test}^{(i)}\right)^2$$

opposed to selecting the model from the **test set**, the model will be scored against the **validation set** for a less biased selection

$$\begin{aligned}
&d = 1 \rightarrow h_\theta(x) = \theta_0 + \theta_1 x \rightarrow \min_\theta J(\theta) \rightarrow \theta^{(1)} \rightarrow J_{cv}(\theta^{(1)})\\
&d = 2 \rightarrow h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 \rightarrow \theta^{(2)} \rightarrow J_{cv}(\theta^{(2)})\\
&d = 3 \rightarrow h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3 \rightarrow \theta^{(3)} \rightarrow J_{cv}(\theta^{(3)})\\
&\vdots \qquad\qquad \vdots\\
&d = 10 \rightarrow h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10} \rightarrow \theta^{(10)} \rightarrow J_{cv}(\theta^{(10)})
\end{aligned}$$

choose $\theta_0 + \theta_1 x + \cdots + \theta_4 x^4$

estimate generalization set error $J_{test}(\theta^{(4)})$

because the parameter *d* is fit to the **validation set** as opposed to the **test set**, the test set can now be used toscore the generaliztion error of the selected model

example

a model selection procedure chooses the degree of polynomial using a cross validation set: for the final model (with parameters θ), it would be generally expected for $J_{cv}(\theta)$ to be lower than $J_{test}(\theta)$ because...

an additional parameter (*d* = degree of polynomial) has been fit to the **cross validation set**