Session #06

CMSC 409: Artificial Intelligence

http://www.people.vcu.edu/~mmanic/

Virginia Commonwealth University,
Fall 2023,
Dr. Milos Manic
(mmanic@vcu. edu)

1

CMSC 409: Artificial Intelligence

Session # 06

Topics for today

- Announcements
- Previous session review
- Normalization
- Classification and Prediction
- Regression Analysis
  - Types, history
  - Least square method
  - Measure of goodness-of-fit
  - Multiple linear, nonlinear, other regression types

2

CMSC 409: Artificial Intelligence
Session # 06

Topics for today (cont.)

- Regression Analysis
  - Accuracy & error measures
    - *Accuracy, misclassification rate, confusion matrix*
    - *Other measures (TP, FP, TN, FN, P)*
    - *Accuracy vs. threshold*
    - *Predictor error measures*

3

CMSC 409: Artificial Intelligence
Announcements Session # 06

- Canvas
  - *New slides posted*
  - *Slide "Things to remember…" added*
  - *2 supplementary files posted (starts with Session_06_ExtraMat...)*
- Office hours zoom
  - *Zoom disconnects me after 45 mins of inactivity. Feel free to chat me via zoom if that happens and I will reconnect (zoom chat welcome outside of office hours as well)!*
- Project #1
  - *Deadline Sep. 14 (noon)*
- Paper (optional)
  - *First draft - due Sep. 12 (noon)*
  - *Think about the topic of your paper and confirm on 1st draft deliverables (class paper instructions)*
- Subject line and signature
  - *Please use [CMSC 409] Last_Name Question*

4

Project 1:

Normalization

- refresher
- when/why needed

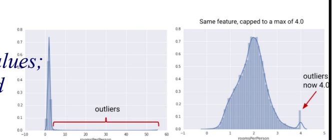
Normalization techniques at a glance (Google ML)

- Original data: $x = (x_1, x_2, \ldots, x_n)$
- Normalized data: $z = (z_1, z_2, \ldots, z_n)$

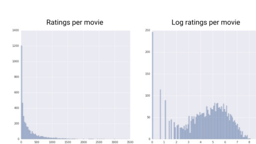
- Scaling to a range (min-max scaling)
  - *To values between 0 & 1; need to know upper/lower bounds; data somewhat uniformly distributed (e.g. age, not income)*

$$z_i = \frac{x_i - \min(x)}{\max(x) - \min(x)}$$

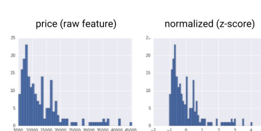
- (Feature) clipping
  - *In case of extreme outlies; caps values outside certain values; clip all temp values above 40 to exactly 40; can be applied before/after other normalizations*



- Log scaling $z_i = \log(x_i)$
  - *When handful values have many points, while most other just a few (e.g. movie ratings – most have few, but a few have lots of ratings)*



- Z-score (standardization) $z_i = (x_i - \mu)/\sigma$
  - *Number of standard deviations* (σ) *away from the mean* (μ)
  - *When you have a few outliers, but not so extreme that clipping is needed; to ensure* $\mu=0$ *and* $\sigma=1$;



Google ML https://developers.google.com/machine-learning/data-prep/transform/normalization

Normalization (min-max scaling)

- Normalizing data to [0-1] range
  - Normalization is used to scale data

$$z_i = \frac{x_i - \min(x)}{\max(x) - \min(x)}$$

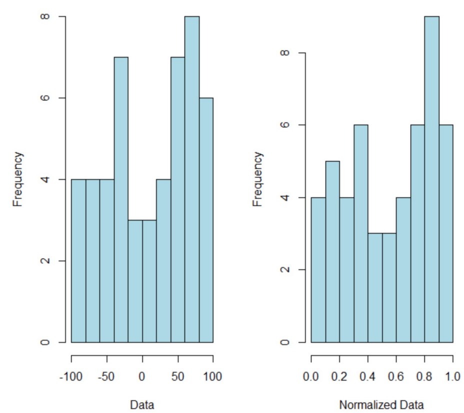
  - Original data:

$$x = (x_1, x_2, \ldots, x_n)$$

  - Normalized data:

$$z = (z_1, z_2, \ldots, z_n)$$

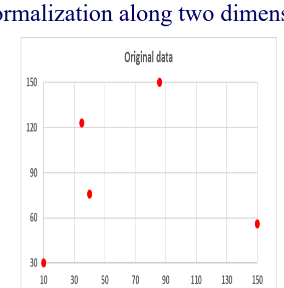
Normalization along one dimension

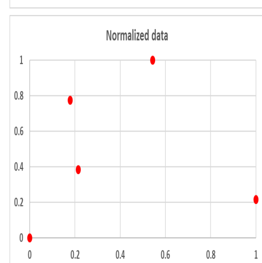


https://stats.stackexchange.com/questions/70801/how-to-normalize-data-to-0-1-range

Normalization (min-max scaling)

Normalization along two dimensions





- Normalizing data to [0-1] range
  - Normalization is used to scale data

$$z_i = \frac{x_i - \min(x)}{\max(x) - \min(x)}$$

  - Original data:

$$x = (x_1, x_2, \ldots, x_n)$$

  - Normalized data:

$$z = (z_1, z_2, \ldots, z_n)$$

Note: sensitive to outliers!

More at: Statology, Wiki statistics, Python sklearn, StackExchange, Google ML

Standardization

(if you have used this in place of normalization, no need to change)

• Standard score (z-score):

$$z_i = \frac{x_i - \mu}{\sigma}$$

• σ – standard deviation of the population, μ - mean of the population, z - distance between x_i and the population mean in units of the standard deviation (z is negative when the x is below the mean, positive when above)

• Normalized data:

$$z = (z_1, z_2, \dots, z_n)$$

Note:

- Standardization creates new data not bounded (unlike normalization); can be negative.
- Normalization usually means to scale a variable to have values between 0 and 1, while standardization transforms data to have a mean of zero and a standard deviation of 1.
- Normalization or standardization, it should be applied to a whole (complete) dataset.

Classification and Prediction

Various Approaches

- Decision trees
- Support Vector Machines
- Neural Networks
  - *Error Back Propagation, Kohonen Winner Take All (WTA) and Self Organizing Maps (SOM), Counter Propagation Networks (CPN), RBF, LVQ*
- Bayesian classification
  - *Naïve Bayesian classification ,belief networks*
- Hard clustering
  - *k-means clustering*
- Learning from neighbors
  - *k-nearest neighbor classifier, Case based reasoning*
- Rule Based Classification
- Fuzzy logic
  - *c-means clustering,*
- Genetic algorithms
- Regression
  - *Linear, non-linear, fuzzy regression*

Classification and Prediction

- Regression Analysis
  - *Types, history*
  - *Least square method*
  - *measure of goodness-of-fit*
  - *multiple linear, nonlinear, fuzzy regression*
  - *Accuracy & error measures*
    - *accuracy, misclassification rate, confusion matrix*
    - *Other measures (TP, FP, TN, FN, P)*
    - *Accuracy vs. threshold*
    - *Predictor error measures*

Review – Regression Analysis

Multiple linear regression

2-dim, 3-dim…(more than one predictor variable)

$$y = w_1 x + w_0;$$

$$y = w_2 x_2 + w_1 x_1 + w_0;$$

$$y = w_3 x_3 + w_2 x_2 + w_1 x_1 + w_0$$

Nonlinear regression

Polynomial regression

• single independent variable (predictor); note that x is not necessarily the perfect predictor of y

$$y = w_3 x^3 + w_2 x^2 + w_1 x + w_0$$

convert to linear form by $x_1 = x,\ x_2 = x^2,\ x_3 = x^3$

$$y = w_3 x_3 + w_2 x_2 + w_1 x_1 + w_0$$

and use least squares method as before….

Review – Regression Analysis

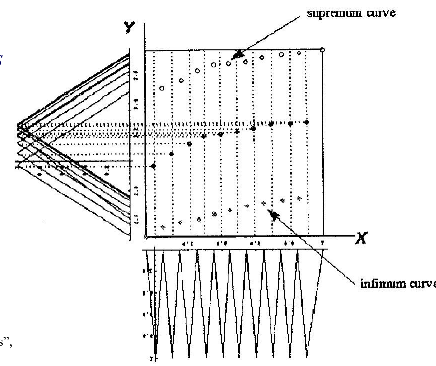
Nonlinear regression

Polynomial & nonpolynomial regression

- *parabola*
- *trigonometric*

Fuzzy regression

- *fuzzy linear regression using least squares*
- *Why?*
  - *data uncertainty drives solution uncertainty*



S. Roychowdhury, "Fuzzy Curve Fitting Using Least Square Principles", Computational Cybernetics, 29 Soft Computing, 1998.

Classification and Prediction

- Regression Analysis
  - *Types, history*
  - *Least square method*
  - *measure of goodness-of-fit*
  - *multiple linear, nonlinear, fuzzy regression*
  - *Accuracy & error measures*
    - *accuracy, misclassification rate, confusion matrix*
    - *Other measures (TP, FP, TN, FN, P)*
    - *Accuracy vs. threshold*
    - *Predictor error measures*

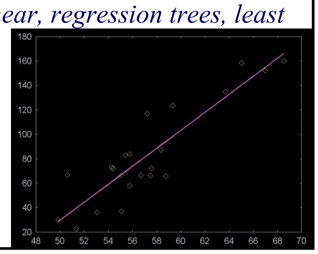
Classification and Prediction

Regression

• *Method for fitting a curve (not necessarily a straight line) through a set of points using some goodness-of-fit criterion. The most common type of regression is linear regression* http://mathworld.wolfram.com/Regression.html

Regression analysis models the predictor-response relationship

- *Independent variable – predictor (known values)*
- *Dependent variable – response (values we are trying to predict)*
- Types
  - *Linear, nonlinear, nonlinear as linear (which one was that?)*
- Curve fitting - examples
  - *Generalized linear, Poisson regression, log-linear, regression trees, least square, spline, fractal*
- Example:
  - *Salary vs. years of experience*

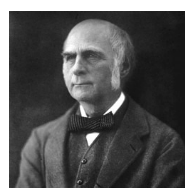


http://www.bearcave.com/misl/misl_tech/wavelets/stat/index.html

Khan Academy, easy to watch videos: https://www.khanacademy.org/math/statistics-probability/describing-relationships-quantitative-data/residuals-least-squares-rsquared/v/regression-line-example

15

Review – Regression Analysis



Regression Analysis

History

- *Introduced by Sir Francis Galton, 18th century*
  - *(cousin of C. Darwin) , known by* ***regression toward the mean****, fingerprint, weather map*
    - ***regression toward the mean:*** *offspring of exceptional individuals tend on average to be less exceptional than their parents (closer to their more distant ancestors – pure statistical reasons)*

About

- *Regression equation demonstrates relation between dependent variable and independent variables*
- *Regression parameters are estimated from set of I/O data*
- *Used for prediction, curve fitting, time-series modeling*

Example $$f(x_i) = w_1 x_i + w_0 + err_i$$

- *Linear regression (can be solved by least square method)*

In statistics, regression toward (or to) the mean is the phenomenon that if a variable is extreme on its first measurement, it will tend to be closer to the average on its second measurement—and, paradoxically, if it is extreme on its second measurement, it will tend to have been closer to the average on its first. To avoid making wrong inferences, regression toward the mean must be considered when designing scientific experiments and interpreting data. http://en.wikipedia.org/wiki/Regression_toward_the_mean

16

Classification and Prediction

- Regression Analysis
  - *Types, history*
  - *Least square method*
  - *measure of goodness-of-fit*
  - *multiple linear, nonlinear, fuzzy regression*
  - *Accuracy & error measures*
    - *accuracy, misclassification rate, confusion matrix*
    - *Other measures (TP, FP, TN, FN, P)*
    - *Accuracy vs. threshold*
    - *Predictor error measures*

17

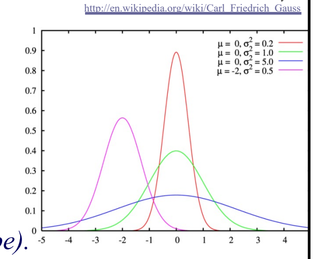
Review – Regression Analysis

Courtesy of:
http://en.wikipedia.org/wiki/Carl_Friedrich_Gauss

Least Square Method

History

- *Introduced by Johann Carl Friedrich Gauss or Gauß , 17th century*
  - *Famous German mathematician and scientist*
  - *Normal (Gaussian) probability distribution*



About

- *Minimizing the sum of squares of errors*

Example

- *For linear regression, minimize TE with respect to the parameters **a** and **b** (or weights w_1 & w_0, interc/slope).*

$$f(x_i) = \hat{y}_i;\ \hat{y}_i \approx y_i,\ \ \hat{y}_i = y_i + err_i,\ \ y_i = w_1 x_i + w_0$$

$$TE = \sum_{i=1}^{n} \left(\hat{y}_i - \left(w_1 x_i + w_0\right)\right)^2 \qquad \sigma^2 = \frac{1}{N}\sum (x_i - \bar{x}_i)^2 \qquad pdf = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)}$$

Linear (closed form), non-linear (iterative – Newton's, grad descent, GN, LM)

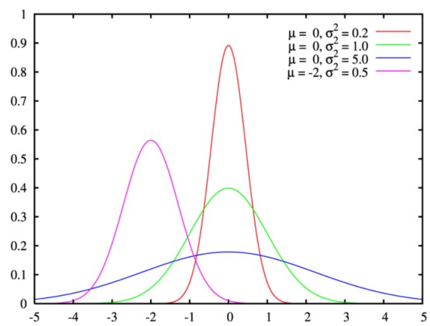
18

Review – Regression Analysis

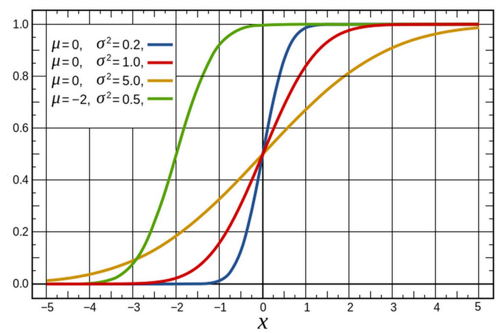
Least Square Method

Standard Normal Distribution

- *If mean=0 and variance = 1, the distribution is called **standard normal distribution** or the **unit normal distribution** denoted by N(0, 1);*
- *A random variable with that distribution is a standard normal deviate.*



probability density function (pdf)



Cumulative distribution function (cdf)

Courtesy of: http://en.wikipedia.org/wiki/Carl_Friedrich_Gauss

Review – Regression Analysis

Least Square Method

Example

- *For linear regression, minimize TE with respect to the weights w_1 & w_0.*

$$TE = \sum_{i=1}^{n}\left(\hat{y}_i - \left(w_1 x_i + w_0\right)\right)^2 \quad or\ simpler\ \ TE = \sum\left(y - \left(w_1 x + w_0\right)\right)^2$$

$$\begin{cases} w_0 = \bar{y} - w_1 \bar{x} \\ w_1 = \dfrac{\sum xy - n\bar{x}\bar{y}}{\sum x^2 - n\bar{x}^2} \end{cases} \quad or \begin{cases} w_0 = \bar{y} - w_1 \bar{x} \\ w_1 = \dfrac{\sum\left(x - \bar{x}\right)\left(y - \bar{y}\right)}{\sum\left(x - \bar{x}\right)^2} \end{cases}$$

after finding derivatives of TE with respect to w1, w0, and setting derivatives to 0…(see derivation by R. Bloom)

http://facultyfiles.deanza.edu/gems/bloomroberta/M10DeriveLeastSquares.doc

*R^2 is a **coefficient of determination** (measure of goodness-of-fit of linear regression), where $\hat{y}$ and y are modelled and original values.*

$$R^2 = \frac{ESS}{TSS} = 1 - \frac{SSR}{TSS}$$

$$SSR = \sum\left(\hat{y} - y\right)^2; TSS = \sum\left(\hat{y} - \bar{y}\right)^2; ESS = \sum\left(y - \bar{y}\right)^2;$$

SSR (TE) - sum of the squared residuals (residual sum of squares)
TSS – total sum of squares
ESS - explained sum of squares (regression sum of squares)

We want to minimize SSR (or TE).

Review – Regression Analysis

after finding derivatives of TE with respect to w1, w0, and setting derivatives to 0…(see derivation by R. Bloom)

We want to minimize the sum of the squared residuals: $\text{SSE} = \sum_{\substack{all\\data}} (y - \hat{y})^2$

But $\hat{y} = a + bx$, so we can substitute into SSE to get $\text{SSE} = \sum_{\substack{all\\data}} (y - a - bx)^2$

Since we want to find the values of a and b that make SSE a minimum, a and b are the variables. Take the derivative of SSE of with respect to a and the derivative of SSE with respect to b. Then set the derivatives equal to 0, to obtain equations which we will later solve to find the values of a and b.

$$\frac{\partial}{\partial a}\sum_{\substack{all\\data}}\left[(y-a-bx)^2\right] = \sum_{\substack{all\\data}}\left[2(y-a-bx)(-1)\right] = -2\sum_{\substack{all\\data}}(y-a-bx) = 0$$

$$\frac{\partial}{\partial b}\sum_{\substack{all\\data}}\left[(y-a-bx)^2\right] = \sum_{\substack{all\\data}}\left[2(y-a-bx)(-x)\right] = -2\sum_{\substack{all\\data}}\left[(y-a-bx)x\right] = -2\sum_{\substack{all\\data}}(xy-ax-bx^2) = 0$$

By breaking up the sums, we can "simplify" this into the two equations with two unknowns a and b

$$-\sum_{\substack{all\\data}} y \;+\; na \;+\; b\sum_{\substack{all\\data}} x \;=\; 0 \qquad\qquad -\sum_{\substack{all\\data}}(xy) \;+\; a\sum_{\substack{all\\data}} x \;+\; b\sum_{\substack{all\\data}}\left(x^2\right) \;=\; 0$$

These equations are linear in a and b, so they are not "difficult" to solve, although the algebra requires a lot of care and patience because the coefficients of the variables a and b are sums. Some cleverness in substituting means for sums helps to further "simplify" the equations to make them easier to work with. Solving these equations to obtain the values of a and b that will minimize the SSE gives us:

$$a = \frac{\sum_{\substack{all\\data}} y \;-\; b\sum_{\substack{all\\data}} x}{n} = \bar{y} - b\bar{x}$$

$$-\sum_{\substack{all\\data}}(xy) \;+\; \left(\bar{y} - b\bar{x}\right)\sum_{\substack{all\\data}} x \;+\; b\sum_{\substack{all\\data}}\left(x^2\right) \;=\; 0$$

$$-\sum_{\substack{all\\data}}(xy) \;+\; \bar{y}\sum_{\substack{all\\data}} x - b\bar{x}\sum_{\substack{all\\data}} x \;+\; b\sum_{\substack{all\\data}}\left(x^2\right) \;=\; 0$$

$$-\sum_{\substack{all\\data}}(xy) \;+\; n\bar{y}\bar{x} - bn\bar{x}\bar{x} \;+\; b\sum_{\substack{all\\data}}\left(x^2\right) \;=\; 0$$

Finally, $b = \dfrac{\sum_{\substack{all\\data}}(xy) \;-\; n\bar{x}\bar{y}}{\sum_{\substack{all\\data}}\left(x^2\right) - n\bar{x}^2}$; after finding b substitute its value to find a using $a = \bar{y} - b\bar{x}$

21

Review – Regression Analysis

after finding derivatives of TE with respect to w1, w0, and setting derivatives to 0…(see derivation by R. Bloom)

We want to minimize the sum of the squared residuals: $\text{SSE} = \sum_{\substack{all\\data}} (y - \hat{y})^2$

But $\hat{y} = a + bx$, so we can substitute into SSE to get $\text{SSE} = \sum_{\substack{all\\data}} (y - a - bx)^2$

Since we want to find the values of a and b that make SSE a minimum, a and b are the variables. Take the derivative of SSE of with respect to a and the derivative of SSE with respect to b. Then set the derivatives equal to 0, to obtain equations which we will later solve to find the values of a and b.

$$\frac{\partial}{\partial a}\sum_{\substack{all\\data}}\left[(y-a-bx)^2\right] = \sum_{\substack{all\\data}}\left[2(y-a-bx)(-1)\right] = -2\sum_{\substack{all\\data}}(y-a-bx) = 0$$

$$\frac{\partial}{\partial b}\sum_{\substack{all\\data}}\left[(y-a-bx)^2\right] = \sum_{\substack{all\\data}}\left[2(y-a-bx)(-x)\right] = -2\sum_{\substack{all\\data}}\left[(y-a-bx)x\right] = -2\sum_{\substack{all\\data}}(xy-ax-bx^2) = 0$$

By breaking up the sums, we can "simplify" this into the two equations with two unknowns a and b

$$-\sum_{\substack{all\\data}} y \;+\; na \;+\; b\sum_{\substack{all\\data}} x \;=\; 0 \qquad\qquad -\sum_{\substack{all\\data}}(xy) \;+\; a\sum_{\substack{all\\data}} x \;+\; b\sum_{\substack{all\\data}}\left(x^2\right) \;=\; 0$$

These equations are linear in a and b, so they are not "difficult" to solve, although the algebra requires a lot of care and patience because the coefficients of the variables a and b are sums. Some cleverness in substituting means for sums helps to further "simplify" the equations to make them easier to work with. Solving these equations to obtain the values of a and b that will minimize the SSE gives us:

22

Review – Regression Analysis

after finding derivatives of TE with respect to w1, w0, and setting derivatives to 0…(see derivation by R. Bloom)

$$a = \frac{\sum_{\substack{all\\data}} y \;-\; b\sum_{\substack{all\\data}} x}{n} = \bar{y} - b\bar{x}$$

$$-\sum_{\substack{all\\data}} (xy) \;+\; (\bar{y} - b\bar{x})\sum_{\substack{all\\data}} x \;+\; b\sum_{\substack{all\\data}} (x^2) = 0$$

$$-\sum_{\substack{all\\data}} (xy) \;+\; \bar{y}\sum_{\substack{all\\data}} x - b\bar{x}\sum_{\substack{all\\data}} x \;+\; b\sum_{\substack{all\\data}} (x^2) = 0$$

$$-\sum_{\substack{all\\data}} (xy) \;+\; n\bar{y}\bar{x} - bn\bar{x}\bar{x} \;+\; b\sum_{\substack{all\\data}} (x^2) = 0$$

Finally, $b = \dfrac{\sum_{\substack{all\\data}} (xy) \;-\; n\bar{x}\bar{y}}{\sum_{\substack{all\\data}} (x^2) - n\bar{x}^2}$; after finding b substitute its value to find a using $a = \bar{y} - b\bar{x}$

Review – Regression Analysis

Least Square Method

*R^2 is a **coefficient of determination** (measure of goodness-of-fit of regression), where **ŷ** is the actual value or data, targeted value (aka predicted, modeled), and **y** is its approximation.*

$$R^2 = \frac{SSR}{TSS} = 1 - \frac{ESS}{TSS}$$

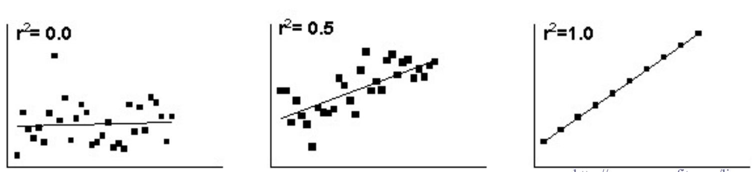
Consult: http://www.stat.ufl.edu/~winner/mar5621/mar5621.doc

$$SSR = \sum(\hat{y} - \bar{y})^2 ; TSS = \sum(y - \bar{y})^2 ; ESS = \sum(y - \hat{y})^2 ;$$

SSR - sum of the squared residuals (residual sum of squares)
TSS – total sum of squares
ESS - explained sum of squares (regression sum of squares)

An R^2 is a value between 0.0 and 1.0.

- ***R^2 = 0.0** means no linear relationship between X and Y (knowing X does not help predict Y); Best-fit line is a horizontal line going through the mean of all Y values.*
- ***R^2 = 1.0** means all points lie exactly on a straight line with no scatter (knowing X lets you predict Y perfectly)*



http://www.curvefit.com/linear_regression.htm

Review – Regression Analysis

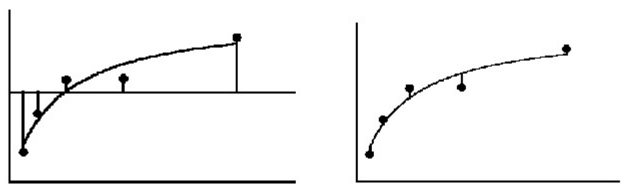
Least Square Method

*R^2 - **coefficient of determination** or measure of goodness-of-fit of regression, where $\hat{y}$ is "desired" output, while y is the "actual" output of our model.*

$$R^2 = \frac{SSR}{TSS} = 1 - \frac{ESS}{TSS}$$

$$SSR = \sum\left(\hat{y} - \bar{y}\right)^2; TSS = \sum\left(y - \bar{y}\right)^2; ESS = \sum\left(y - \hat{y}\right)^2;$$

An R^2 is a value between 0.0 and 1.0, and describes the discrepancies between expected and modeled values.



http://www.graphpad.com/curvefit/goodness_of_fit.htm

Classification and Prediction

- Regression Analysis
  - *Types, history*
  - *Least square method*
  - *measure of goodness-of-fit*
  - *multiple linear, nonlinear, fuzzy regression*
  - *Accuracy & error measures*
    - *accuracy, misclassification rate, confusion matrix*
    - *Other measures (TP, FP, TN, FN, P)*
    - *Accuracy vs. threshold*
    - *Predictor error measures*

Accuracy & Error Measures

Accuracy estimation techniques

- ***Accuracy or recognition rate*** *of classifier M performed on* ***test*** *patterns:*

$$ACC(M) = \frac{correctly\_classified\_patterns}{total\_set\_of\_patterns} \qquad \text{(\#right/\#total)}$$

- ***Error (misclassification) rate*** *of M estimated on* ***testing*** *set:*

$$1 - ACC(M)$$

- ***Resubstitution error*** *– estimated on* ***training*** *set (the error rate on the training data)*
- ***Confusion matrix***
  - *for m classes of mxm dimension; should have zeros outside of main diagonal;*
  - *class i (row) labeled by classifier as class j (column)*

Classes	*buys_computer = yes*	*buys_computer = no*	*Total*	*Recognition (%)*
buys_computer = yes	6,954	46	7,000	99.34
buys_computer = no	412	2,588	3,000	86.27
Total	7,366	2,634	10,000	95.52

Accuracy & Error Measures

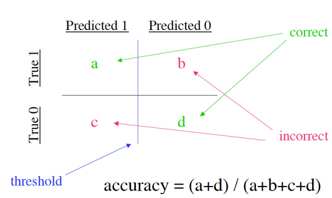
Classes	*buys_computer = yes*	*buys_computer = no*	*Total*	*Recognition (%)*
buys_computer = yes	6,954	46	7,000	99.34
buys_computer = no	412	2,588	3,000	86.27
Total	7,366	2,634	10,000	95.52

Accuracy estimation techniques

- ***For positive/negative patterns***

	Predicted 1	Predicted 0
Actual 1	True positives (a)	False negatives (b)
Actual 0	False positives (c)	True negatives (d)

Confusion Matrix



accuracy = (a+d) / (a+b+c+d)

True positives

- *when actual or real, system value is 1, and predictor also "predicts" 1*

http://www.cs.cornell.edu/courses/cs678/2006sp/performance_measures.4up.pdf
Thorsten Joachims, CS6780 Advanced ML, https://www.cs.cornell.edu/people/tj/

Accuracy & Error Measures

Classes	buys_computer = yes	buys_computer = no	Total	Recognition (%)
buys_computer = yes	6,954	46	7,000	99.34
buys_computer = no	412	2,588	3,000	86.27
Total	7,366	2,634	10,000	95.52

Accuracy estimation techniques

- ***For positive/negative patterns***

	Predicted 1	Predicted 0
Actual 1	True positives (a)	False negatives (b)
Actual 0	False positives (c)	True negatives (d)

- ***If accuracy = 97% but only 3% are actual positives? Other measures are needed.***
  - *ACC (accuracy), Recall or True Positive rate (TP), False Positive rate (FP), True Negative rate (TN), False Negative rate (FN), Precision (P):*

$$ACC(M) = \frac{a+d}{a+b+c+d} \quad TP(M) = \frac{a}{a+b} \quad FP(M) = \frac{c}{c+d}$$

$$TN(M) = \frac{d}{c+d} \quad FN(M) = \frac{b}{a+b} \quad P(M) = \frac{a}{a+c}$$

http://www2.cs.uregina.ca/~dbd/cs831/notes/confusion_matrix/confusion_matrix.html
http://www.cs.cornell.edu/courses/cs678/2006sp/performance_measures.4up.pdf

Accuracy & Error Measures

Classes	buys_computer = yes	buys_computer = no	Total	Recognition (%)
buys_computer = yes	6,954	46	7,000	99.34
buys_computer = no	412	2,588	3,000	86.27
Total	7,366	2,634	10,000	95.52

Accuracy estimation techniques

- ***For positive/negative patterns***

	Predicted 1	Predicted 0
Actual 1	True positives (a)	False negatives (b)
Actual 0	False positives (c)	True negatives (d)

- *ACC (accuracy),* the proportion of the total number of predictions that were correct
- *True Positives (TP),* is the proportion of positive cases that were correctly identified
- *False Positive rate (FP),* proportion of negatives cases that were incorrectly classified as positive
- *True Negative rate (TN),* proportion of negatives cases that were classified correctly
- *False Negative rate (FN),* proportion of positives cases that were incorrectly classified as negative
- *Precision (P),* proportion of the predicted positive cases that were correct:

$$ACC(M) = \frac{a+d}{a+b+c+d} \quad TP(M) = \frac{a}{a+b} \quad FP(M) = \frac{c}{c+d}$$

$$TN(M) = \frac{d}{c+d} \quad FN(M) = \frac{b}{a+b} \quad P(M) = \frac{a}{a+c} \; (also \; \frac{d}{b+d})$$

http://www2.cs.uregina.ca/~dbd/cs831/notes/confusion_matrix/confusion_matrix.html
http://www.cs.cornell.edu/courses/cs678/2006sp/performance_measures.4up.pdf

Accuracy & Error Measures

Accuracy estimation techniques

- *For positive/negative patterns*

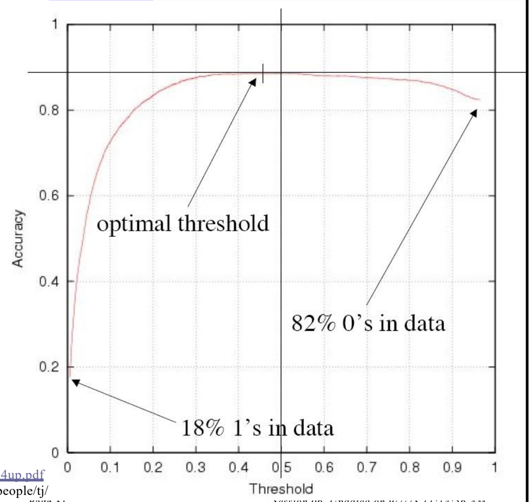
	Predicted 1	Predicted 0
Actual 1	True positives (a)	False negatives (b)
Actual 0	False positives (c)	True negatives (d)

- *If max(f(x)) < Threshold*
  - *all cases predicted 0s*

	Predicted 1	Predicted 0
Actual 1	0 (a)	False negatives (b)
Actual 0	0 (c)	True negatives (d)

- *If min(f(x)) > Threshold*
  - *all cases predicted 1s*

	Predicted 1	Predicted 0
Actual 1	True positives (a)	0 (b)
Actual 0	True negatives (c)	0 (d)



http://www.cs.cornell.edu/courses/cs678/2006sp/performance_measures.4up.pdf
Thorsten Joachims, CS6780 Advanced ML, https://www.cs.cornell.edu/people/tj/

Accuracy & Error Measures

Percent reduction in error (marketing?)

- *Example:*
  - *80% accuracy = 20% error*
  - *suppose learning increases accuracy from 80% to 90%*
  - *error reduced from 20% to 10%*
  - *50% reduction in error*

- *if learning increases accuracy…*
  - *99.90% to 99.99% = 90% reduction in error (error from 0.10 to 0,01)*
  - *50% to 75% = 50% reduction in error*
  - *Can be applied to many other measures*

http://www.cs.cornell.edu/courses/cs678/2006sp/performance_measures.4up.pdf

Accuracy & Error Measures

Accuracy estimation techniques

- *For positive/negative patterns*

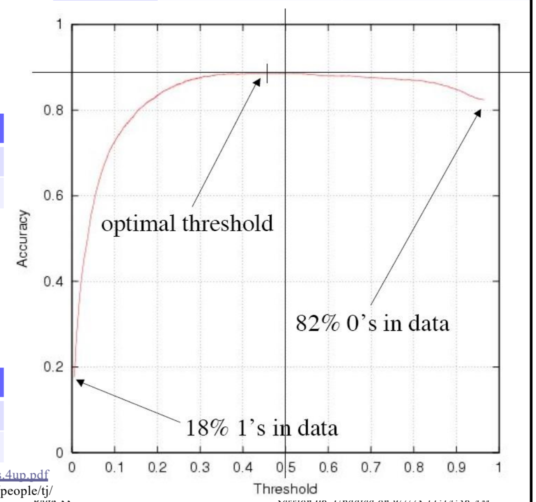
	Predicted 1	Predicted 0
Actual 1	True positives (a)	False negatives (b)
Actual 0	False positives (c)	True negatives (d)

- *If max(f(x)) < Threshold*
  - *all cases predicted 0s*

	Predicted 1	Predicted 0
Actual 1	0 (a)	False negatives (b)
Actual 0	0 (c)	True negatives (d)

- *If min(f(x)) > Threshold*
  - *all cases predicted 1s*

	Predicted 1	Predicted 0
Actual 1	True positives (a)	0 (b)
Actual 0	True negatives (c)	0 (d)



http://www.cs.cornell.edu/courses/cs678/2006sp/performance_measures.4up.pdf
Thorsten Joachims, CS6780 Advanced ML, https://www.cs.cornell.edu/people/tj/

33

Regularization

- *Over/underfitting,*
- *accuracy vs. generalization*

very briefly, details in deep learning spring (grad course)…

References:

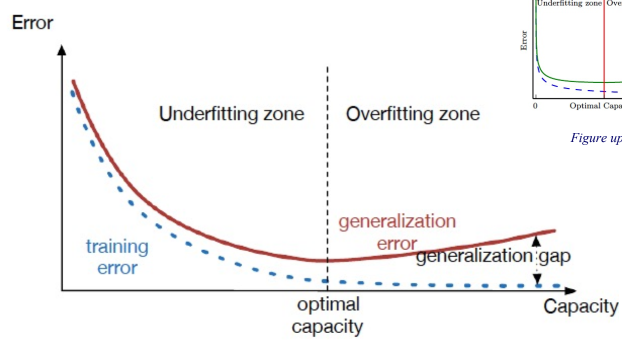
1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning*, MIT Press, url=http:/ www.deeplearningbook.org, year=2016 (Ch.5 ML Basics, Ch.7 Regularization in DL).

34

Regularization for Deep Learning

Regularization

- How do these relate? (Ch.5)
  - ***training error, testing error, over/under fitting, and capacity***



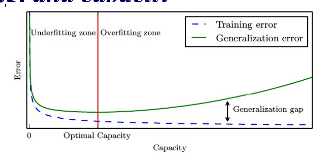


Figure updated in later edition…

Low capacity – underfitting zone; as we increase capacity, training error decreases, but the gap between training and generalization error increases; later on, the size of this gap outweighs the decrease in training error (overfitting zone).

Things to remember…

- **True positives vs. true positive rates**
  - *True positives, negatives…are "absolute" instances/patterns (a, b, c, d in matrix); their rates (TP rate, FP rate) are not (are ratios)!*
- **Values in confusion matrix - which are more important?**
  - *True positives and true negatives are paramount to predict correctly*
  - *False negatives – dangerous (predictor missed an event)*
  - *False positives (leading to distrust in predictor)*
- **Least Square Method**
  - *for linear regression, minimize total error (TE) (square of errors)*
  - *TE is typical metric (stopping criterion) for training of neural networks*
- **Overtraining…**
  - *Very small error not always "good"*
  - *We try to balance accuracy (small TE) and ability to generalize*
  - *Hard to predict; typical when model starts "seeing" new data from the system (i.e. when too late, model is in production)*