ENSF 444: Machine Learning Systems

Week 3 – Linear Models

UNIVERSITY OF CALGARY

Lecture Goals

UNIVERSITY OF CALGARY

- Introduce linear models for regression and classification

- Introduction to Machine Learning with Python
  - Chapter 2 Section 2.3.3 Linear models (p.47)
  - Chapter 2 Section 2.3.3 Linear models for classification (p.58)
- An Introduction to Statistical Learning with Applications in Python
  - Chapter 3 Section 1

Supervised Learning

Review: What is supervised learning?

- Supervised learning is used whenever we want to predict a certain outcome from a given input, and we have examples of input/output pairs
  - The output is typically referred to the class or label of the data
- We can build a machine learning model from these input/output pairs, given by the training set
- Our goal is to make accurate predictions for new, never-before-seen data

Review: Types of supervised learning

- There are two major types of supervised machine learning problems, called **classification** and **regression**
- In **classification**, the goal is to predict a **discrete** class label, which is a choice from a predefined list of possibilities
- For **regression**, the goal is to predict a **continuous** number. The predicted value can be any number within a given range

Advertising Example

- You have been hired as a consultant for a marketing firm, that is trying to figure out how much money to spend on different types of advertising to increase sales
- You have been given historical data on the following:
  - TV advertising budget
  - Radio advertising budget
  - Newspaper advertising budget
  - Sales

Advertising Example

UNIVERSITY OF CALGARY

- You want to answer the following questions:
  - Is there a relationship between advertising budget and sales?
  - Which media are associated with sales?
  - How accurately can we predict future sales?
  - Is the relationship linear?
- Linear models can be used to answer these questions

Linear Models

What are linear models?

- Linear models are supervised learning algorithms that predict an output variable based on a linear combination of input features
- They can be used for both regression and classification tasks, depending on whether the output variable is continuous or categorical

Some Common Linear Models

- **Linear regression:** Predicts a continuous output variable from one or more input features
  - For example, it can model how the price of a house depends on the size, location and other factors
- **Logistic regression:** Predicts a binary output variable from one or more input features
  - For example, it can estimate the probability of a patient having a heart disease or not
- Linear models are simple, interpretable, and fast to train. However, they may not perform well on complex or non-linear data.

Linear Models for Regression

What is linear regression?

- Simple linear regression assumes there is a linear relationship between the input (x) and the output (y)
- Where:

$$y \approx \beta_0 + \beta_1 x$$

- For example, x may represent TV advertising and y may represent sales:

$$sales \approx \beta_0 + \beta_1 (TV)$$

What is linear regression?

- On the previous slide, β_0 and β_1 are two unknown constants that represent the **intercept** and **slope** of the linear model
- Together, they are referred to as the **model coefficients**
- We can use the training data to estimate $\widehat{\beta_0}$ and $\widehat{\beta_1}$
- We can then use the estimated coefficients to predict future sales based on a new value for TV advertising:

$$\hat{y} = \widehat{\beta_0} + \widehat{\beta_1}x$$

Solving for the model coefficients

UNIVERSITY OF CALGARY

- We want to find an intercept and slope that represent a linear trend that is as close as possible to the data
- The most common approach is to minimize the **least squares** criterion
- The least squares approach chooses the model coefficients that minimize the residual sum of squares (RSS) or mean squared error (MSE)
- MSE = RSS / number of samples

Solving for the model coefficients

- We define RSS as:

$$RSS = {e_1}^2 + {e_2}^2 + {e_3}^2 + \cdots + {e_n}^2$$

- Where:

$$e_i = y_i - \widehat{y_i}$$

- This represents the ith residual (difference between predicted and true values)

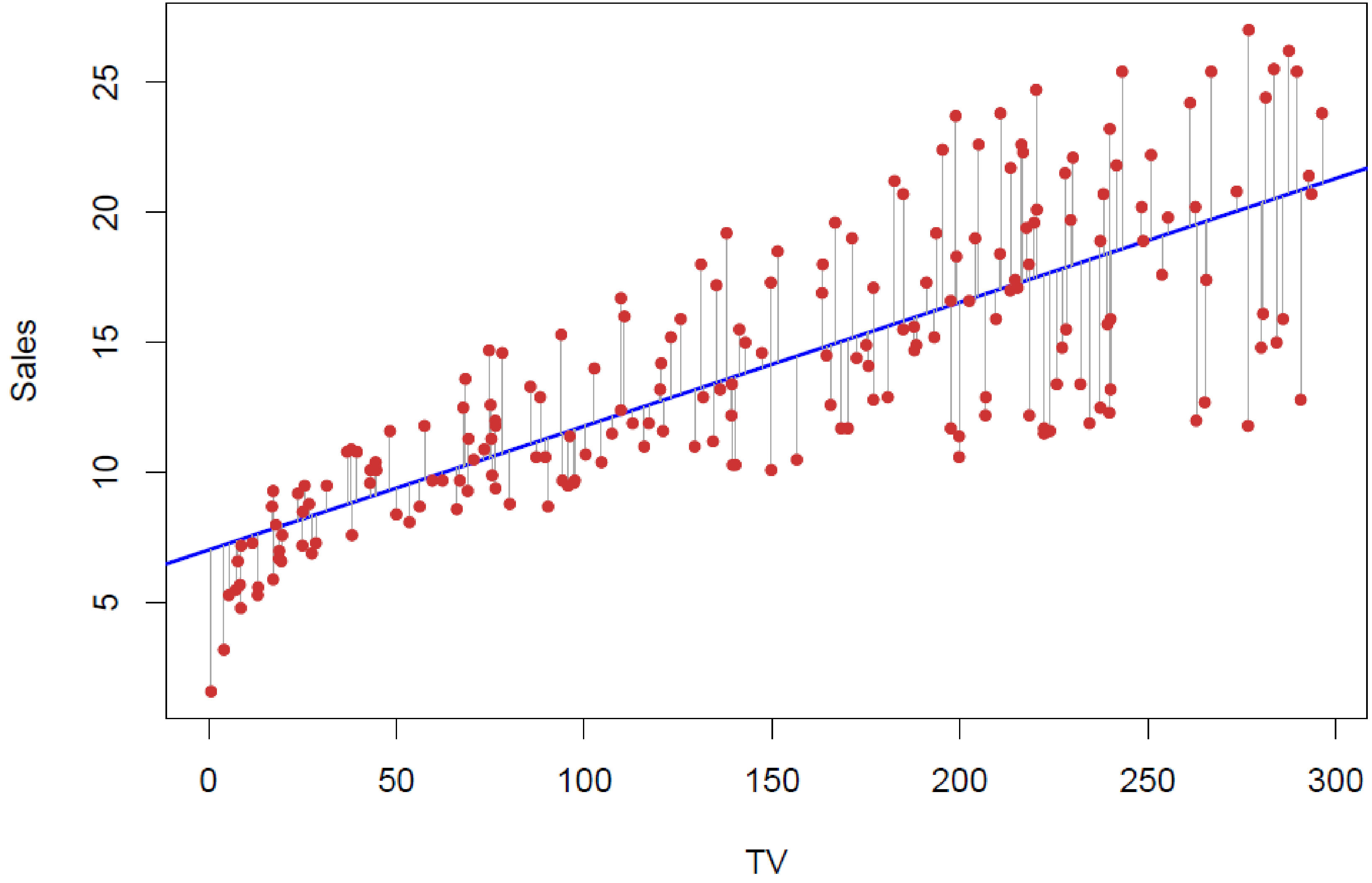

FIGURE 3.1. *For the* `Advertising` *data, the least squares fit for the regression of* `sales` *onto* `TV` *is shown. The fit is found by minimizing the residual sum of squares. Each grey line segment represents a residual. In this case a linear fit captures the essence of the relationship, although it overestimates the trend in the left of the plot.*

Multiple linear regression

UNIVERSITY OF CALGARY

- For cases with multiple features, the general prediction formula for a linear model looks as follows:

$$\hat{y} = w[0] * x[0] + w[1] * x[1] + ... + w[p] * x[p] + b$$

- Here, x[0] to x[p] denotes the features (in this example, the number of features is p+1) of a single data point, w and b are parameters of the model that are learned, and $\hat{y}$ is the prediction the model makes

Multiple linear regression (cont'd)

- For a dataset with a single feature, this is:

$$\hat{y} = w[0] * x[0] + b$$

- This is the equation for a line
- Here, w[0] is the slope and b is the y-axis offset (intercept)
- For more features, w contains the slopes along each feature axis
- Alternatively, you can think of the predicted response as being a weighted sum of the input features, with weights (which can be negative) given by the entries of w

Multiple linear regression (cont'd)

- There are many different linear models for regression
- The difference between these models lies in how the model parameters (w and b) are learned from the training data, and how model complexity can be controlled
- Popular models used:
  - Linear regression (ordinary least squares)
  - Ridge regression
  - Lasso regression

Linear Regression

- Also known as ordinary least squares (OLS)
- This is the simplest linear method for regression
- Linear regression finds the parameters w and b that minimize the **mean squared error** between predictions and the expected values for the training set
- The mean squared error is the sum of the squared differences between the predictions and the true values, divided by the number of samples
- Linear regression has no parameters, which is a benefit, but it also has no way to control model complexity

Ridge Regression

UNIVERSITY OF CALGARY

- Ridge regression is also a linear model for regression, so it uses the same formula as ordinary least squares
- For ridge regression, the coefficients (w) are not only chosen so that they predict well on the training data, but so they can also fit an additional constraint
- The additional constraint is that the magnitude of coefficients must be as small as possible; all entries of w should be close to zero

Ridge Regression (cont'd)

- Having the coefficients close to zero means each feature should have as little effect on the outcome as possible (which translates to having a small slope), while still predicting well
- This constraint is an example of what is called **regularization**
- **Regularization** means explicitly restricting a model to avoid overfitting
- Ridge regression uses **L2 regularization**

Lasso Regression

- An alternative to Ridge for regularizing linear regression is **Lasso**
- As with ridge regression, the lasso also restricts coefficients to be close to zero, but in a slightly different way, called **L1 regularization**

Lasso Regression (cont’d)

UNIVERSITY OF CALGARY

- The consequence of L1 regularization is that when using the lasso, some coefficients are exactly zero
- This means some features are entirely ignored by the model
- This can be seen as a form of automatic feature selection
- Having some coefficients be exactly zero can make a model easier to interpret and can reveal the most important features of your model

Other models

- scikit-learn also provides the ElasticNet class, which combines the regularization of Lasso and Ridge
- In practice, this combination works best, though you will need to adjust two parameters: one for the L1 regularization, and one for the L2 regularization

Regression accuracy metrics

- https://scikit-learn.org/stable/modules/model_evaluation.html#regression-metrics
    - **R^2 score (coefficient of determination)**
    - Mean absolute error (MAE)
    - **Mean squared error (MSE)**
    - Mean squared logarithmic error (MSLE)
    - Mean absolute percentage error (MAPE)
    - Median absolute error (MedAE)
    - And many more…

Linear Models for Classification

Linear models for classification

- Linear models are also extensively used for classification
- Let's look at an example for binary classification. In this case, a prediction is made using the following formula:

$$\hat{y} = w[0] * x[0] + w[1] * x[1] + \ldots + w[p] * x[p] + b > 0$$

- The formula looks very similar to the one for linear regression, but instead of returning the weighted sum of the features, we threshold the predicted value at zero

Linear models for classification (cont'd)

- If the function is smaller than zero, we predict that class is –1
- If it is larger than zero, we predict the class is +1
- This prediction rule is common to all linear models for classification
- There are many ways to find the coefficients (w) and the intercept (b)

Linear models for classification (cont’d)

- There are many algorithms for learning linear models
- These algorithms all differ in the following two ways:
  - The way in which they measure how well a particular combination of coefficients and intercept fits the training data
  - If and what kind of regularization they use
- The two most common linear classification algorithms are:
  - Logistic Regression
  - Linear Support Vector Machines (SVM)

Multiclass classifier

UNIVERSITY OF CALGARY

- Many linear classification models are for binary classification only, and don't extend naturally to the multiclass case
  - Except for logistic regression
- A common technique to extend a binary classification algorithm to a multiclass classification algorithm is the one-vs.-rest approach

Multiclass classifier (cont’d)

- In the one-vs.-rest approach, a binary model is learned for each class that tries to separate that class from all the other classes, resulting in as many binary models as there are classes
- To make a prediction, all binary classifiers are run on a test point
- The classifier that has the highest score on its single class “wins”, and this class label is returned as the prediction

Classification accuracy metrics

- https://scikit-learn.org/stable/modules/model_evaluation.html#classification-metrics
    - Accuracy score
    - Confusion matrix
    - F1 score
    - And many more…

Strengths, Weaknesses and Parameters

Linear model parameters

- The main parameter of linear models is the regularization parameter, called alpha in the regression models and C in the classification models
- Large values for alpha or small values for C represent simple models
- Tuning these parameters is quite important
- Typically, C and alpha are searched for on a logarithmic scale

Linear model parameters (cont'd)

UNIVERSITY OF CALGARY

- The other decision you must make is whether you want to use L1 regularization or L2 regularization
- If you assume that only a few of your features are important, you should use L1, otherwise, you should default to L2
- L1 can also be useful if interpretability of the model is important. As L1 will use only a few features, it is easier to explain which features are important to the model, and what the effects of these features are

Linear model strengths

- Linear models are very fast to train and fast to predict
- They scale to very large datasets and work well with sparse data
- Linear models make it relatively easy to understand how a prediction is made, using the previous formulas for regression and classification
- Linear models often perform well when the number of features is large compared to the number of samples

Linear model weaknesses

- Unfortunately, it is often not entirely clear why coefficients are the way they are
- This is particularly true if your dataset has highly correlated features; in these cases, the coefficients might be hard to interpret
- They are also often used on very large datasets, simply because it’s not feasible to train other models
- However, in lower-dimensional spaces, other models might yield better generalization performance