Carl Morgenstern

EE598 Machine Learning for Power Systems, HW2

Problem 1:

1. Artificial intelligence describes when a computer can do something in a “smart” or human way. Such as playing a chess game. Machine learning is a subset of AI, but the program can determine the constraints of the problem by using data rather than having them determined by a human. Deep learning is a subset of Machine learning that uses Neural Networks, that is particularly useful for image or speech recognition. The deep, refers to deeper layers of inputs that the neural network creates, for facial image recognition, layer 1: pixels; layer 2: edges of color; layer 3: facial features (eyes, nose, mouth, hair); layer (4) determine if the object has a face.
2. My background comes from Power. I have worked at a utility for the past year and a half, where my priority has been building distribution circuit models and organizing data. I have used distribution circuit models to analyze effects of PV, run fault studies, protection coordination studies, and voltage level studies. I mainly organize data from distribution circuit maps in AutoCAD, using Visual Basic scripts to modify the maps so that I can extract data accurately. Though the way I have been modifying the circuit maps is accurate, because the maps are often poorly drawn and there is ambiguous data, it takes time to modify the maps so that the data is organized and can be extracted. I know that machine learning could be used here.
3. Utilities know surprisingly little about their distribution system. Even if utilities have access to AMI meters, they don’t have a way to analyze the data the meters are collecting. One of the uses could be to identify behind-the-meter devices such as home-based batteries, or EVs. Using AMI data, a machine learning program could be trained with customers that are known to have these devices.
4. A successful application of machine learning in the power domain has been uses google maps satellite images to identify solar panels. By training a dataset with a wide range of solar panel satellite images, a utility can search their service territory to identify locations of installed PV. Based on the size of the solar panel array, the size can be estimated.

Problem 2:

1. Bernoulli distribution can only have two outcomes, 0 or 1 with different weights assigned to the possibility of each outcome. Binomial distribution is the sum of independent and identical Bernoulli actions. Example: Bernoulli is a coin flip with odds of getting head. Binomial is a 4 coin flips, with odds of getting 0,1,2,3,4 heads.
2. Binomial has a limited number of trails, and therefore a limited number of possible outcomes. Poisson has an unlimited number of possible outcomes, and is a probability distribution. Binomial distribution example: there are a 100, words on a page, each word has a 1% chance of having a typo, plot the outcomes are 0 to 100 words having a typo. Poisson distribution example: on average there is 4 typos per page in a book, what is the probability there are 20 on a page? The poission distribution does not need to specify the number of words, (maybe each page is infinite words), but the further you get from the mean, the probability of that action occurring decreases. The poisson distribution can describe the binomial distribution as the number of trails reach infinity.

Source: https://keydifferences.com/difference-between-binomial-and-poisson-distribution.html

1. The exponential distribution is the probability distribution that describes the time or distance between Poisson events. So there are on average 4 typos per day, a poisson distribution, the distance on a page between typos can be described by an exponential distribution with an theta = ¼.

Source: <https://www.youtube.com/watch?v=FvsuFa0PkG4>

1. Law of large numbers is a theorem says that if you preform the same experiment many times, the average result should be close to the expected value, and the average will be more accurate to the expected value as the trails of the same experiment is increases.

Source: <https://en.wikipedia.org/wiki/Law_of_large_numbers>

1. No matter what the natural distribution of a system is, as your sample size approaches infinity, the distribution of frequency of the means of the system will begin to be perfectly represented by a normal distribution.

Source : https://www.khanacademy.org/math/ap-statistics/sampling-distribution-ap/sampling-distribution-mean/v/central-limit-theorem

1. The sum of independent squared normal distributions, is a chi-squared distribution. What a chi-squared statistic informs us is how much difference exists between the two distributions. A low value for chi-square means there is a high correlation between the two sets of data. This can be used to calculate correlation between two sets of normal distributions or test to see how well the expected distribution matches with the input data. As the independent normal variables increase (or the degrees of freedom), the chi-squared distribution approaches a normal distribution.

Source: <http://www.statisticshowto.com/probability-and-statistics/chi-square/>

1. We must be able to understand how humans have traditionally described data sets, and the tools they have used to analyze them. This allows us to describe what the machine learning programs we are designing are going to do in terms of traditional techniques. Anyone without any understanding of data can throw it into a machine learning program and generate results but they will have no way of explaining how they got from the inputs to the outputs. By understanding the distributions, we can extract approximate parameters from the data and mathematically prove what our machine learning program is doing, and that it follows natural trends.

Problem 3:

2. The dataset contains cases from a study that was conducted between 1958 and 1970 at the University of Chicago's Billings Hospital on the survival of patients who had undergone surgery for breast cancer.

4. There are 4 input variables

5.

Variance Age: 116.71458266366656

Mean Age: 52.45751633986928

Variance Op\_year: 10.558630665380907

Mean Op\_year: 62.85294117647059

Variance ax\_nodes: 51.69111753991214

Mean ax\_nodes: 4.026143790849673

Variance survival: 0.19527483124397302

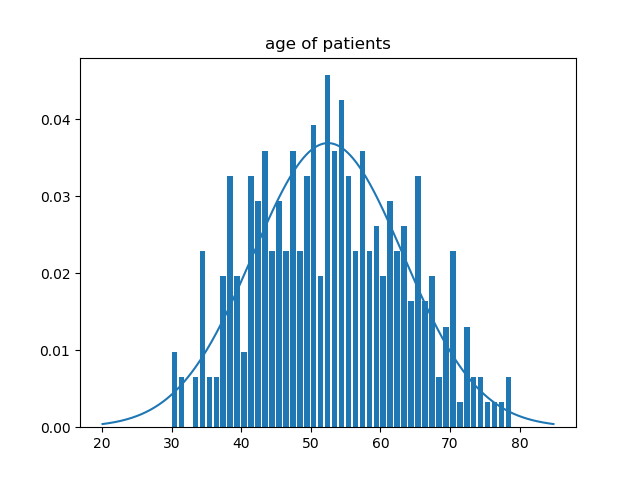
Mean survival: 1.2647058823529411

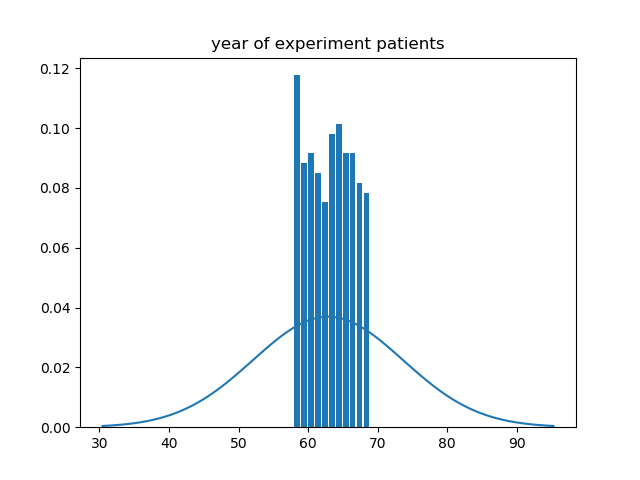
6. Math, handwritten on a sheet. Correlation coefficient p = 0.08952944559093895

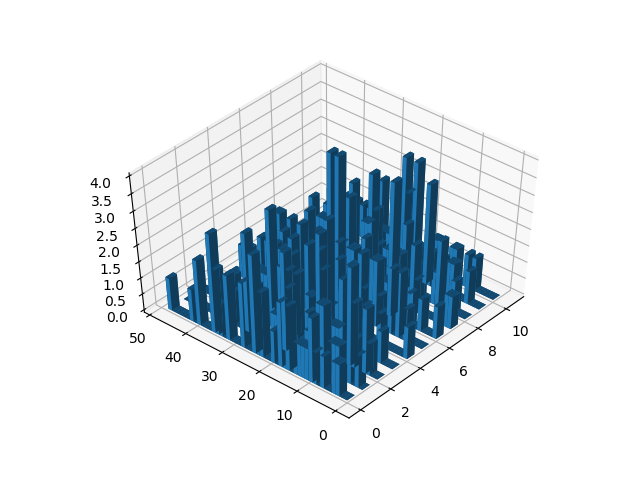
7. marginalized op year. variance: 0.00014058307022551543 mean: 0.09090909090909091

Once marginalized, they are not the same as the original variance and mean.

8. The data for the first variable (age of patients) fits ok, but there is still a lot of error. The second variable (experiment per year) fit poorly



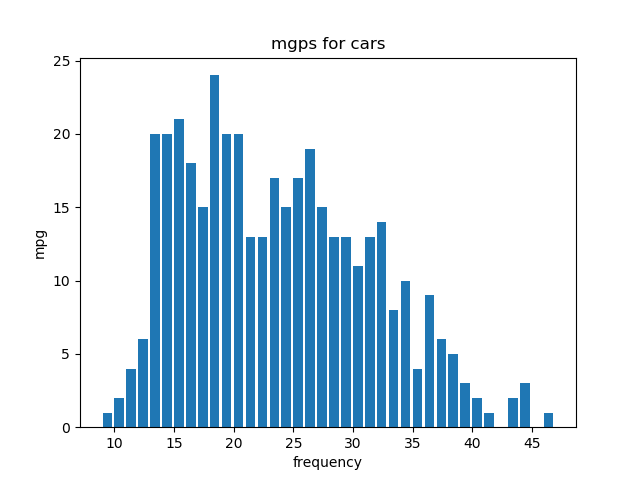


9. There seems to be almost no correlation between the two variables, this was confirmed by a very low correlation coefficient.

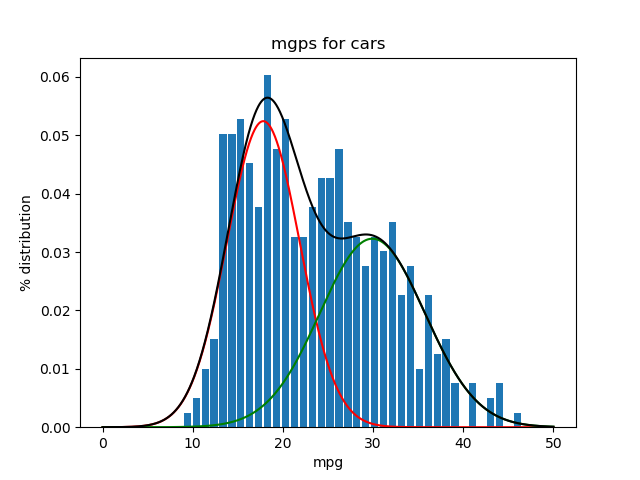
Problem 4:

1. Gaussian mixture distribution is a combination when multiple gaussian distribution are combined to describe one variable. When data has more than 1 peak.
2. The integral over the whole space must be 1.
3. The dataset is mpg of cars. The variable I will be plotting over is MPG of different cars.

4.



5. I will plan on using two bases. Im going to fit the data to a guassian mixture function using python. I’m estimating that the mean of the bases will be around (18, 28). To estimate the variance of the two peaks, I’d draw normal distribution curves around both peaks. I would try to guess the standard deviation by estimating how far from the mean encapsulates ~1/3 of the data. Then I would square estimated standard deviation to find the variance. The weights would be calculated by roughly totaling up the data under both guassians. In the above dataset I’m guessing that the weights will be about 50% for both curves.



The means ended up being: [17.88, 29.995]

The variance: [16.517, 33.518]

The weights: [0.535, 0.465]

Problem 5a:

1.

Dataset: <https://data.nrel.gov/submissions/69>

Residential load profiles, and generated electric vehicle load profiles.

This dataset is separated into 3 files. Generated customer profiles for 200 homes for a year. Level 1 charging (1920W) EV generated profiles for 348 different EVs. Level 2 charging (6600W) EV generated profiles for 348 different EVs. The profiles are for **a whole year at 10-minute intervals**. There are ~54,000 datapoints for a whole year. The household data was extracted from a larger dataset: <https://www.eia.gov/consumption/residential/data/2009/index.php?view=microdata>

2.

Inputs: household, car, charging level, time, day. Ouput: Electricity consumption

3.

Assessing impact of charging on a grid. Forecasting the charging.

4.

You can infer roughly the next time the vehicle is going to charge based on previous charging patterns.

States of charging the vehicle. If it is charging, or not charging.

5.

I would like to do a project on identifying if customers have EVs or batteries at their home. So, I would create a set of training data, by adding the profiles that correlate together to each other. I would see how this training data correlates to each other. Do an FFT on the EV profiles to determine if there are times that have strong correlations. Determine slopes between, datapoints to see if there is a correlation between charging and the change of total consumption. Basically, try to identify prominent features and correlations to determine new features that aren’t the starting variables.

Problem 6:

1.

Filter method looks at each variable independently and determines weights of those variables correlating to the data. The highest weighted variables are then used. The wrapper method looks at sets of variables to see which set can describe the data best, so the variables are not looked at independently, and the highest ranked set of variables are used.

2.

In the power domain there are a bunch of applications that would be useful for machine learning. I would use the filter method first to determine the weights of correlation from my inputs to outputs so I can better understand the relation of each variable to my dataset. But then I would probably use the wrapper method. This tends to be more accurate and even if two variables are correlated to the output, they might also be strongly correlated to each other, and the wrapper method can do a better job at seeing the whole spectrum, rather than just one aspect of the output.

3.

The dataset is an archive of spam emails. There are 48 continuous attributes of the frequency of certain words in the email. 6 attributes for frequency of characters in the email, the average length of uninterrupted sequence of capital letters, length of longest uninterrupted sequence of capital letters, total number of capital letters in the email, and then if the email was considered spam or not.

4.

Source: <https://machinelearningmastery.com/feature-selection-machine-learning-python/>

with open(sys.path[0]+"\\spam\_data.csv") as f:

df = pandas.read\_csv(f)

arr = df.values

#arr = numpy.around(arr,3)

spam\_data = arr[:,0:57]

spam\_out = arr[:,57]

# feature extraction

test = SelectKBest(score\_func=chi2, k=10,)

fit = test.fit(spam\_data,spam\_out)

#pandas.set\_option('display.float\_format', lambda x: '%.3f' % x)

# summarize scores

numpy.set\_printoptions(precision=3, suppress=True)

print(fit.scores\_)

ind = fit.get\_support(True)

print(ind)

5.

I used the a chi2 best fit filter method. It calculates the weight of the features correlating to the output. In the above scenario, I chose to extract the 10 best features, and their indices are stored in the “ind” array. The weights are calculated by the chi2 test, the higher the correlation, the higher the weights.

Problem 7:

1.

Same as problem 5.

<https://data.nrel.gov/submissions/69>

2.

I will be adding the PEV consumption to the households. My outputs will be the electricity consumption of the vehicle, because that is what I am going to try to predict, based on the following inputs: Time (hour and minute), Time (day of the week), total Watts, change in total Watts. Unlike the Spam dataset in problem 6, there are not a lot of inputs per output, and if I use this dataset for future research, I will have to come up with other inputs, such as time from last charge. I could also organize the data into different subsets to determine if different correlations between work days and weekends, or during hours of the day.

Other Features I could generate but are not explicitly stated in the data: change of slope from last 20-minute charge, change (slope) of electric consumption from last hour, electric consumption difference from previous day at same time, electric consumption difference from previous week at same day and time.

3.

with open(sys.path[0]+"\\combined\_EV\_HH.csv") as f:

df = pandas.read\_csv(f)

df['Time'] = pandas.to\_datetime(df['Time'])

df['Time'] = [d.hour\*60 + d.minute for d in df['Time']]

arr = df.values

print (arr)

#arr = numpy.around(arr,3)

data = arr[:,[0,3]]

EVcharge = arr[:,1]

# feature extraction

test = SelectKBest(score\_func=chi2, k='all',)

fit = test.fit(data,EVcharge)

#pandas.set\_option('display.float\_format', lambda x: '%.3f' % x)

# summarize scores

numpy.set\_printoptions(precision=3, suppress=True)

print(fit.scores\_)

features = fit.transform(data)

# time (minute of the day) Total Electricity Consumption

weights = [ 339287.492 23670375.544]

4.

In the excel file, I added the EV consumption of vehicle 1 to a household 1 consumption to create a total. Then I imported the dataset into python. I extract the minute of the day from the dataset. My input array (data) was minutes, and total consumption. My output data array (EVcharge) was the charging data from the EV. Then I used a chi2 to weight the two inputs to the output. The was correlation to both the inputs was then printed out.

It seems that there is a stronger correlation to the sum of the electricity than to the minute of the day, but the weights are both very strong.

I wanted to use the change in consumption, but chi2 is only for non-negative values. I plan on using this data for my project, and will spend more time creating and analyzing other inputs (mentioned in question 5), and use many different feature selection methods. It would also be interesting to test other households, to see how if the time feature changes.