BINF 4008 / COMS 4995 Assignment #1

David Davila-Garcia (dmd2225)

Due: 11:59 PM, Friday, September 29th, 2023

1 Logistic Regression and Gradient Descent (30 Points)

1. **(3 Points + 1 Bonus Point) Explain the difference between a generative and discriminative classifier. Is logistic regression an example of a generative or discriminative model? What about naïve Bayes? (Bonus question: What makes naïve Bayes "naïve?")**

   Generative Classifier: class of models that simulates the overall distribution P(X,Y) and applies Bayesian inference to find $P(Y|X)$ from the estimated P(X,Y) distribution. Naive Bayes is an example of a generative model and is "naive" because it makes the assumption that each feature x_i is independent of all other features $x_j \in x|x_{j\neq i}$, and all features are equally important in producing the outcome y.

   Discriminative Classifier: class of models which estimate the probability distribution $P(Y|X)$ directly from the data. Logistic Regression is an example of a discriminative model that estimates the values of $P(Y = 1|X)$ and $P(Y = 0|X)$ from the training data by minimizing the negative log loss (see below).

2. **(5 Points) We want to learn a logistic regression model to classify whether a patient is at risk of sepsis. Derive the log-loss minimization for the sepsis classification problem using the idea that we can model binary labels $P(y = 1|\mathbf{x}) \sim$ Bernoulli$(\sigma(\mathbf{w}^T\mathbf{x}))$, where $\sigma(\mathbf{x}) = \frac{1}{1+e^{-\mathbf{x}}}$ is the sigmoid function.**

$$P(y = 1|\mathbf{x}) \sim \text{Bernoulli}(\frac{1}{1 + e^{-\mathbf{w}^T\mathbf{x}}})$$

$$P(y = 1|\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T\mathbf{x}}}$$

$$1 - P(y = 1|\mathbf{x}) = P(y = 0|\mathbf{x}) = \frac{1}{1 + e^{\mathbf{w}^T\mathbf{x}}}$$

$$Maximize : Likelihood = \prod_{i=1}^{n} P(\hat{y_i} = 1|\mathbf{x_i})^{y_i} P(\hat{y_i} = 0|\mathbf{x_i})^{1-y_i}$$

$$Maximize : \log(Likelihood) = \sum_{i=1}^{n} \log(P(\hat{y}_i = 1|\mathbf{x_i})^{y_i} P(\hat{y}_i = 0|\mathbf{x_i})^{1-y_i})$$

$$Minimize : -\log(Likelihood) = -\sum_{i=1}^{n} (y_i) \log(P(\hat{y}_i = 1|\mathbf{x_i})) + (1 - y_i) \log(P(\hat{y}_i = 0|\mathbf{x_i}))$$

3. **(5 Points) Now we're ready to train the model. We will use gradient descent to learn w. Get the derivative of $\sigma(w^T x)$ with respect to $w_j \in w$.**

$$t = w^T x$$
$$\sigma(t) = \frac{1}{1 + e^{-t}}$$
$$\frac{\delta \sigma(t)}{\delta t} = \sigma(t) * (1 - \sigma(t))$$
$$\frac{\delta(t)}{\delta w_j} = x_j$$
$$\frac{\delta \sigma(w^T x)}{\delta w_j} = \frac{\delta \sigma(t)}{\delta(t)} * \frac{\delta(t)}{\delta w_j}$$
$$\frac{\delta \sigma(w^T x)}{\delta w_j} = \sigma(w^T x)(1 - \sigma(w^T x))(x_j)$$

4. **(5 Points) Derive a gradient descent update to optimize for w.**

   For one sample, the cost function is:

$$NNL(y, \sigma(w^T x)) = -y \log(\sigma(w^T x)) - (1 - y) \log(1 - \sigma(w^T x))$$

   Calculate derivative of NLL with respect to $w_j$for gradient update:

$$\frac{\delta NLL(y, \sigma(w^T x))}{\delta w_j} = \frac{\delta NLL(y, \sigma(w^T x))}{\delta \sigma(w^T x)} * \frac{\delta \sigma(w^T x)}{\delta w_j}$$

   Solving for $\frac{\delta NLL(y, \sigma(w^T x))}{\delta \sigma(w^T x)}$, given $\frac{\delta}{\delta x} ln(x) = \frac{1}{x}$:

$$\frac{\delta NLL(y, \sigma(w^T x))}{\delta \sigma(w^T x)} = -\frac{y}{\sigma(w^T x)} + \frac{1 - y}{1 - \sigma(w^T x)}$$

   Plug in $\frac{\delta \sigma(w^T x)}{\delta w_j}$ from previous question:

$$\frac{\delta NLL(y, \sigma(w^T x))}{\delta w_j} = -\left[\frac{y}{\sigma(w^T x)} - \frac{1 - y}{1 - \sigma(w^T x)}\right] \sigma(w^T x)(1 - \sigma(w^T x))(x_j)$$
$$= \left[\sigma(w^T x) - y\right] x_j$$

Gradient Descent Update:

$$w_j := w_j - \alpha * \frac{\delta NLL(y, \sigma(w^T x))}{\delta w_j}$$

$$w_j := w_j - \alpha * \left[\sigma(w^T x) - y\right] x_j$$

5. **(8 Points, Programming) Get the Iris dataset from sklearn.datasets. Implement gradient descent for logistic regression with L1 regularization using the base numpy package and an 80:20 train:test split. Plot the loss and ROC curves and report the AUROC and accuracy score.**

   See .ipynb file

6. **(4 Points, Programming) Run the model for at least 5 additional regularization parameters. Plot the weights learned with respect to the regularization parameter using matplotlib.pyplot.stem. Discuss which parameter you will use for the Iris dataset and why?**

   See .ipynb file

2 MIMIC-IV Exploratory Data Analysis (30 Points)

1. **(2 Points) Briefly describe the MIMIC-IV data collection procedure.**

   MIMIC-IV (v2.2) is a retrospective data set from Beth Israel Deaconess Medical Center and composed of deidentified electronic medical records of patients admitted to the ER or ICU between 2008 and 2019. This data set is divided into Hospital and ICU modules:

   hosp: contains hospital-wide EMR for patients admitted to the ER and ICU.

   - admissions: contains demographic information from the patient (insurance, age, race), admission data (time admitted, admission ID), and information regarding if the patient died (hospital expire flag)
   - labevents: all laboratory measurements for patients, including the result, test name, reference ranges, abnormal result indicator, and priority status
   - patients: contain information regarding the patient's gender, day of death, and age at admission

   icu: contains ICU clinical information system data.

   - icustays: has information related to the specific stay (subject_id, stay_id), the length of the stay in the ICU in fractional days, and the time admitted into and out of ICU.
   - chartevents: contains all patient charts during ICU stay. This includes vitals

2. **(3 Points) For Question 3, we want to train a model to predict 48-hour in-hospital mortality for patients admitted to the ICU with lengths of stay for at least 24 hours. All input features will contain data from the first 24 hours only. Define inclusion and exclusion criteria for creating a cohort for this task.**

   Inclusion Criteria: ICU patients admitted to the ICU with lengths of stay >= 24 hours, and unique stay_id's will be the input. The same patient may have multiple stay_id's.
   Exclusion Criteria: No patients were excluded from the original data set if they met the inclusion criteria. Diagnosis ICD codes were not considered, because no datetime information regarding when the diagnosis was made could be parsed from the tables. As such, there would be information leakage since diagnoses could have been made after the ICU stay.

   Variables Considered: must have lab/vital values for 80% of patients

   - age
   - gender
   - race
   - insurance
   - heart rate
   - blood pressure
   - temperature
   - priority - if the lab result is urgent
   - abnormal - if the result is abnormal
   - flag

3. **(10 Points, Programming) Build your study cohort using your cohort definition from question 2.1. Visualize histograms for class prevalence, patient age, gender, insurance, racial identity, blood pressure, and oxygen levels. (Note: we expect you to use more features than these values in your feature vectors).**

4. **(5 Points, Programming) Remove outlier patients based on out-of-range values. Some resources you may want to explore include:**

   - **Tables for acceptable ranges for physiological variables.**
   - **Prior work on ML-based ICU mortality prediction.**

5. **(5 Points) Explain what you did for question 2.4 (i.e. describe the thresholds and procedure you used to remove outliers).**

   Patients were not removed based on out of range results. The ref_range_lower and ref_range_upper variables are the bounds for normal lab values. However, abnormal lab values should not be excluded because they have predictive ability for our task. As such, I removed highly improbable values such as negative values and set an upper bound for measurements.

6. **(5 Points) Generate a table one for your cohort (in LATEX). (You may find pandas.DataFrame.to_latex() useful.)**

Table 1: ICU Stays - Patient Characteristics Stratified by 48-Hour Mortality

	Overall	48-hour Mortality: False	48-hour Mortality: True
N	57 734.00	56 818.00	916.00
Unique hadm_id	[53034]	[52933]	[101]
Unique subject_id	[42264]	[42026]	[238]
Age	65.00(1634)	64.90(1634)	71.19(1526)
Gender			
Male (0)	56.5%	56.6%	52.0%
Female (1)	43.5%	43.4%	48.0%
Race			
WHITE	67.9%	68.0%	61.0%
UNKNOWN	10.9%	10.7%	21.2%
BLACK	10.4%	10.5%	8.4%
OTHER	4.2%	4.2%	3.6%
HISPANIC	3.7%	3.7%	2.5%
ASIAN	2.9%	2.9%	3.3%
Insurance			
Other	46.9%	47.1%	39.4%
Medicare	45.8%	45.7%	55.0%
Medicaid	7.3%	7.3%	5.6%

7. **(Bonus Question, 2 Points) What are some limitations of deploying a model trained on MIMICIV to other clinical settings? (Hint: take a look at this paper on MIMIC-III).**

   One limitation is that the patient population might look very different than other patient populations. As such, models trained on this specific population might not be generalizable to other datasets/populations.

3 48-Hour In-Hospital Mortality Prediction (40 Points)

1. **(2 Points) Describe the utility of a predictive scoring system for adverse outcomes in the intensive care unit. Look up such a (non-ML-based) instrument used in clinical practice. Describe the data collected and how the risk score is calculated.**

In an ICU setting, doctors must routinely perform cost-benefit analysis; with limited time, they balance the needs of patients according to severity: patients perceived to have treatable but life-threatening illnesses are prioritized. Doctors must perceive all the patient's information (vitals, history, symptoms, etc.) in order to assess the severity of their illness. For ICU patients with adverse outcomes, reducing the time to treatment has the ability to save lives. As such, a predictive scoring system for adverse outcomes has the ability to simplify clinician burden by alerting them sooner so they can treat their patients faster.

One non-ML-based instrument used in clinical practice is the Framingham Risk Score, which was compiled from the famous Framingham Heart study in 1948, in which 6000 of the 10000 adults in Framingham MA were enrolled in a longitudinal study to find risk factors for cardiovascular disease (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4159698/). The Risk score is used to calculate an individuals 10 year cardiovascular risk. Per (https://www.mdcalc.com/calc/38/framingl risk-score-hard-coronary-heart-disease), the risk score is calculated as follows:

2. **(2 Points) Give three examples of how data leakage may emerge while training predictive systems in clinical settings.**

   Data leakage may occur if a patient exists in both the training and validation sets. In this case, the model has already seen this patient at a later point chronologically during training, which can allow the model to know if they are alive during testing. Another classic example would be using prescriptions ordered as a feature in a sepsis prediction model; the model would perform well if it can see that antibiotics were ordered in patients with sepsis. Lastly, leakage may occur if you perform standardization or scaling based on weights from the entire dataset instead of only using the training set to derive the scaling factors.

3. **(10 Points, Programming) Create a static feature vector per ICU stay based on the patient history. As mentioned in question 2.2, all input features will be from data collected within 24 hours of hospital admission.**

4. **(3 Points, Programming) Create a train (80%) and test (20%) split to prevent above mentioned data-leakages.**

5. **(5 Points) Describe the procedure you used for the feature vector generation (e.g. we can consider average blood pressure over 24 hours, did you remove missing values or impute, etc.) and data splitting steps (particularly on which unique identifier you split on). Describe the normalization procedure you used for all features.**

   For labs, I used regex to impute values from the "comment" field into the "valuenum" field. I thought this was appropriate because I saw an inverse relationship in the missingness between these two fields. In addition, for the same patient, I computed the mean for the same lab ("item_id") if the lab was repeated multiple times over the 24 hours. I also computed the average "abnormal" tag for labs. I repeated the process for the vitals data, but was unable to impute any values because there was no "comment" column. In addition, I included the average "flag" and "priority" labels for vitals data per patient over the 24 hours. I did not

include icd codes, because I was unable to locate a "time of diagnosis" variable. This would be necessary in order to prevent leakage in our model, since we have the constraint that the variables may only include the first 24 hours during an ICU stay. Values were normalized based on the standard scaler using the training set.

6. **(10 Points, Programming) Train and tune an XGBoost model. Consider which parameters to tune (XGBoost documentation on the topic) and experiment with the scale_pos_weight parameter using 5-fold cross-validation.**

7. **(8 Points) Evaluate the Accuracy, AUROC, AUPRC, Positive-Predictive Value with the 95% confidence intervals and make a table for your results (again, in LATEX). Bold your most performant model based of 5-fold CV. Justify the choice of classification metric you used for model selection.**

   The

Table 2: Model Evaluation Metrics

Metric	Mean	Std
Fit Time	17.941201	0.392995
Score Time	0.080000	0.001000
Test Accuracy	0.984139	0.000679
Test ROC AUC	0.522141	0.008549
Test PR AUC	0.031618	0.011197
Test PPV	0.334947	0.141804

8. **(Bonus Question, 2 Points) How would you improve on the XGBoost model? You can either change feature representations or the model type, or both. Describe and justify your choices.**

   I would change the model to an LSTM and the input features to include time-series data. I am unsatisfied with the performance of my model, because it only uses the mean for every lab/vital measurement grouped by "item_id". It seems unfair to only use 1 number to emulate the trends of an entire distribution.