CS 229 Project Report
George Liu
TA: Albert Haque

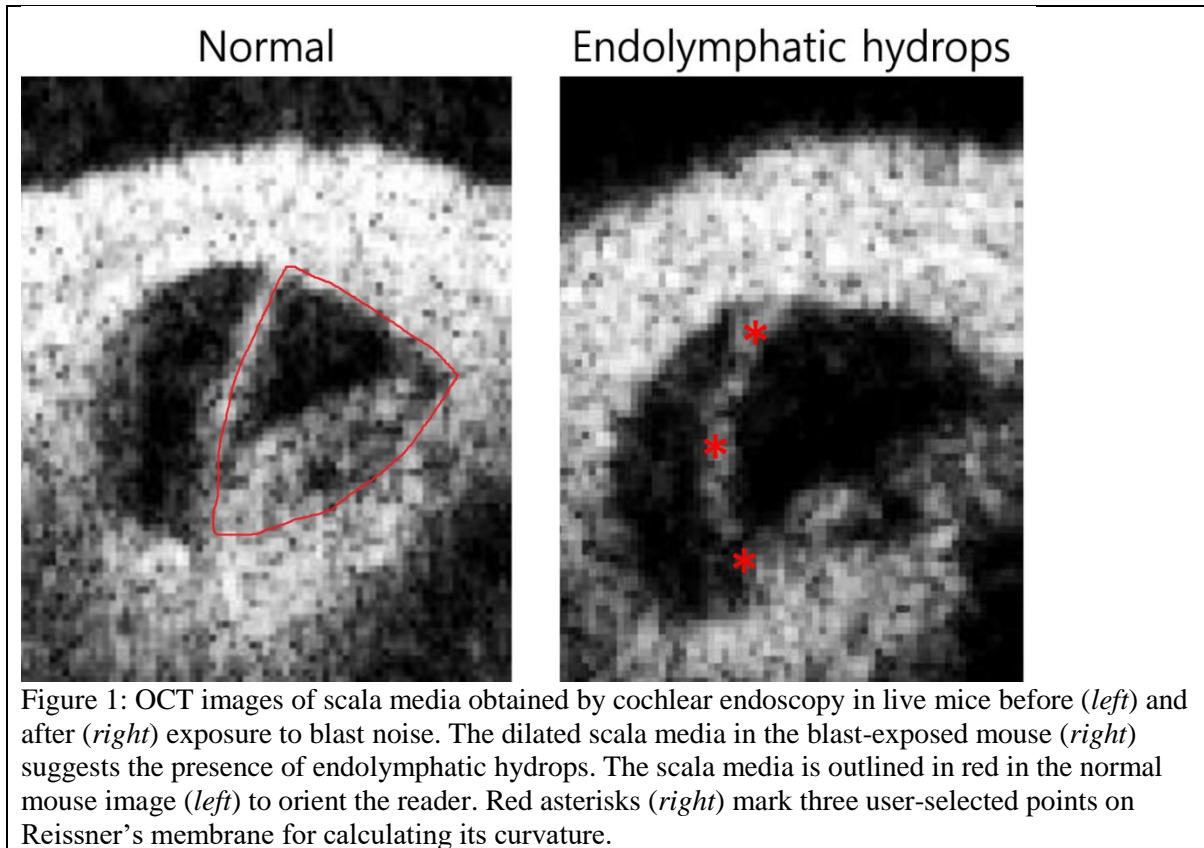**Computer aided detection of endolymphatic hydrops to aid the diagnosis of Meniere's disease**

1. Introduction
Chronic dizziness afflicts over 30% of elderly in the United States, yet remains difficult to manage because of the dearth of sensitive and specific diagnostic tests to differentiate its many potential causes [1,2,3]. Treatment typically involves multiple workups and consultations but often no conclusive diagnosis is reached. Meniere's disease is associated with chronic dizziness, tinnitus, and pressure in the ears due to endolymphatic hydrops, fluid swelling in the scala media compartment inside the cochlea [4]. Although detection of endolymphatic hydrops is the gold standard for diagnosing Meniere's disease, this test currently can only be made during postmortem histopathologic examination [5]. There is no standard method currently to non-invasively detect endolymphatic hydrops in a live patient to reach a conclusive diagnosis of Meniere's disease.

Optical coherence tomography is a new frontier for *in vivo* detection of cochlear abnormalities. Optical coherence tomography (OCT) is routinely applied in non-invasive retina examinations, and has demonstrated potential for producing high-resolution images non-invasively of cochlea in mice [6]. As optical coherence tomography is translated to medical applications for imaging inside the cochlea of humans, machine learning has the potential to aid the classification of OCT images of cochlea as demonstrating endolymphatic hydrops for diagnosing Meniere's disease. Using OCT images of cochlea from healthy mice and mice exposed to large blast sounds to induce endolymphatic hydrops [7], we trained and tested machine learning algorithms to detect endolymphatic hydrops. This is feasible because endolymphatic hydrops is obvious visually in OCT images (Figure 1). **The input to our algorithm is an image and its manually-measured scala media volume. We then use logistic regression, Naïve-Bayes, and SVM to output a predicted classification of endolymphatic hydrops due to blast exposure.**

Section 2: Related Work
Previously, machine learning has been applied to attempt to help make diagnoses based on OCT images in eyes, skin, and coronary arteries [8-12]. In skin, researchers have applied the SVM algorithm to classify skin cancers other than melanoma (actinic keratosis lesions and basal cell carcinomas) using 14 manually measured features of skin from OCT images such as the thickness and number of skin layers [8]. Researchers have also applied SVM to classify stents as covered or not covered using manually annotated OCT images with the goal of assessing delay in arterial healing due to drug-eluting stent implantation [9]. Most work on medical OCT image machine learning has focused on machine learning predictions from eye OCT images using manually measured features of the eye, such as the thickness of the optic nerve and retina thickness, to try to detect early signs of glaucoma [10-13]. These studies use logistic regression and SVM as well as generative learning algorithms to try to classify OCT images of eyes with glaucoma. Although all of these attempts have demonstrated at least partial improvement to detecting signs of disease, they all are still not fully automated, requiring non-trivial manual feature extraction from OCT images, feature selection, and significant amounts of image segmentation (e.g. to detect the retinal area of interest). Moreover none of the techniques have demonstrated sufficient sensitivity and specificity to be routinely applied in clinical decision-making so far. No work has been reported on OCT image analysis of cochlea, since cochlear OCT imaging is rather new [6].

Figure 1: OCT images of scala media obtained by cochlear endoscopy in live mice before (*left*) and after (*right*) exposure to blast noise. The dilated scala media in the blast-exposed mouse (*right*) suggests the presence of endolymphatic hydrops. The scala media is outlined in red in the normal mouse image (*left*) to orient the reader. Red asterisks (*right*) mark three user-selected points on Reissner's membrane for calculating its curvature.

Section 3: Dataset and Features

Data was obtained from OCT imaging of cochlea in 10 distinct mice. Five mice were treated with exposure to blast sound in a custom built noise chamber to induce endolymphatic hydrops [7], and compared with five control of the same age and background. OCT images were taken in mice one, two, and three hours after exposure. A total of 30 OCT images was used for training and cross-validation.

Images of cochlea were obtained as z-stacks of cross-sections of the cochlea. The volume of scala media, curvature of Reissner's membrane bounding the scala media, and time of image acquisition were used as the three features for images. Volumes of the scala media (figure 1) were manually measured in each z-stack file using Amira visualization software prior to this work. These volume measurements were measured along the first 0.15 mm of the basilar membrane.

The curvature of Reissner's membrane (second feature) was calculated as part of the machine learning training software in MATLAB. Each input training image z-stacked was displayed at the middle image of the stack. This layer was chosen because it empirically was determined to capture the scala media cross-section in all images. Three points on Reissner's membrane were specified by the user manually, and from those three coordinates the curvature of Reissner's membrane was calculated using the known mathematical solution for the radius of a circle passing through three points [14]. (Figure 1.)

4. Methods

The machine learning algorithms were coded in MATLAB. The three features and known feature labels (1=exposed to blast, 0=control) were used to train and test the following models: logistic regression, Naïve-Bayes, and support vector machines (SVM) of linear, quadratic, and cubic kernels.

For logistic regression, I used Newton's method as coded in homework 1 with maximum iteration of 30 for calculating theta. This iteration limit was found to be best empirically while working with the training data. Newton's method for estimating the parameters θ uses the formula

(1) $\theta := \theta - H^{-1}\nabla_\theta\ell(\theta)$

Where $\ell(\theta)$ is the log likelihood of θ, H is its Hessian with respect to θ, and the likelihood of θ is calculated using the probability interpretation that $P(y = 1 \mid x; \theta) = h_\theta(x)$, where $h_\theta(x)$ is the logistic function of the linear combination of parameters θ and feature vector x ($\theta_0 + \theta_1x_1 + \theta_2x_2 + \theta_3x_3$) and y is the output variable (1=blast, 0=control). By iterating Eq. (1), Newton's method provides a way to do "gradient ascent" in getting θ to converge to the value that maximizes $\ell(\theta)$. The number of iterations used was 30. Iterations less than 30 such as 10 and 20 also produced fairly good testing results; values above 40, however, produced matrix singularity errors in calculating the Hessian and its inverse.

For the support vector machines (SVMs), built-in MATLAB functions were used to solve the dual optimization problem for maximizing the geometric margin of the support vector decision boundary. The dual optimization is given by

$$\max_\alpha \quad W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} y^{(i)}y^{(j)}\alpha_i\alpha_j\langle x^{(i)}, x^{(j)}\rangle.$$
$$\text{s.t.} \quad \alpha_i \geq 0, \quad i = 1, \dots, m$$
$$\sum_{i=1}^{m} \alpha_iy^{(i)} = 0,$$

where the SVM training function uses sequential minimal optimization to simultaneously update pairs of alpha values to solve the optimization problem and determine the best decision boundary. Intuitively, the SVM training algorithm solves for the decision boundary that maximizes the distance between the boundary and its closest points. Since these closest points therefore determine the decision boundary (regardless of the position of all farther points), these points are known as the support vectors for the SVM decision boundary. Training an SVM is easy as long as it involves calculations only of the inner product of the feature vectors. The SVM can be expanded to include linear, quadratic, and cubic feature mappings (or any other feature mapping) so long as the transformation merely changes the inner product of the feature vectors to an easy-to-calculate inner product of the feature mappings, known as the kernel of the transformation.

Naïve Bayes presupposes that each discretized feature obeys a multinomial Bernoulli probability distribution and that the features are conditionally independent of each other. I used MATLAB's Naïve Bayes function to train and test this model.

5. Experiments/Results/Discussion

The settings used for the support vector machine training were the default MATLAB settings. Since Newton's method was used instead of gradient ascent, a value of alpha (rate of ascent) for maximizing the log likelihood was not needed or specified. The iteration cap was 30 for Newton's method.

Because the data set was small (10 unique mice, five per class, 30 unique observations), training was performed using leave-one-out cross-validation (LOOCV). The percent of data that when left out during training of the model resulted in that datum's misclassification was used to "test" the learning algorithms. In the future, the training and testing data sets should be kept separate during training and cross-validation. Although all models yielded zero training error but some cross-validation error, overfitting was likely not the issue because of the simplicity of the models used, but further work could test this.

Figure 2. Trained models and LOOCV confusion matrices and ROC curves for two learning algorithms.

| Logistic regression | SVM (linear kernel) |
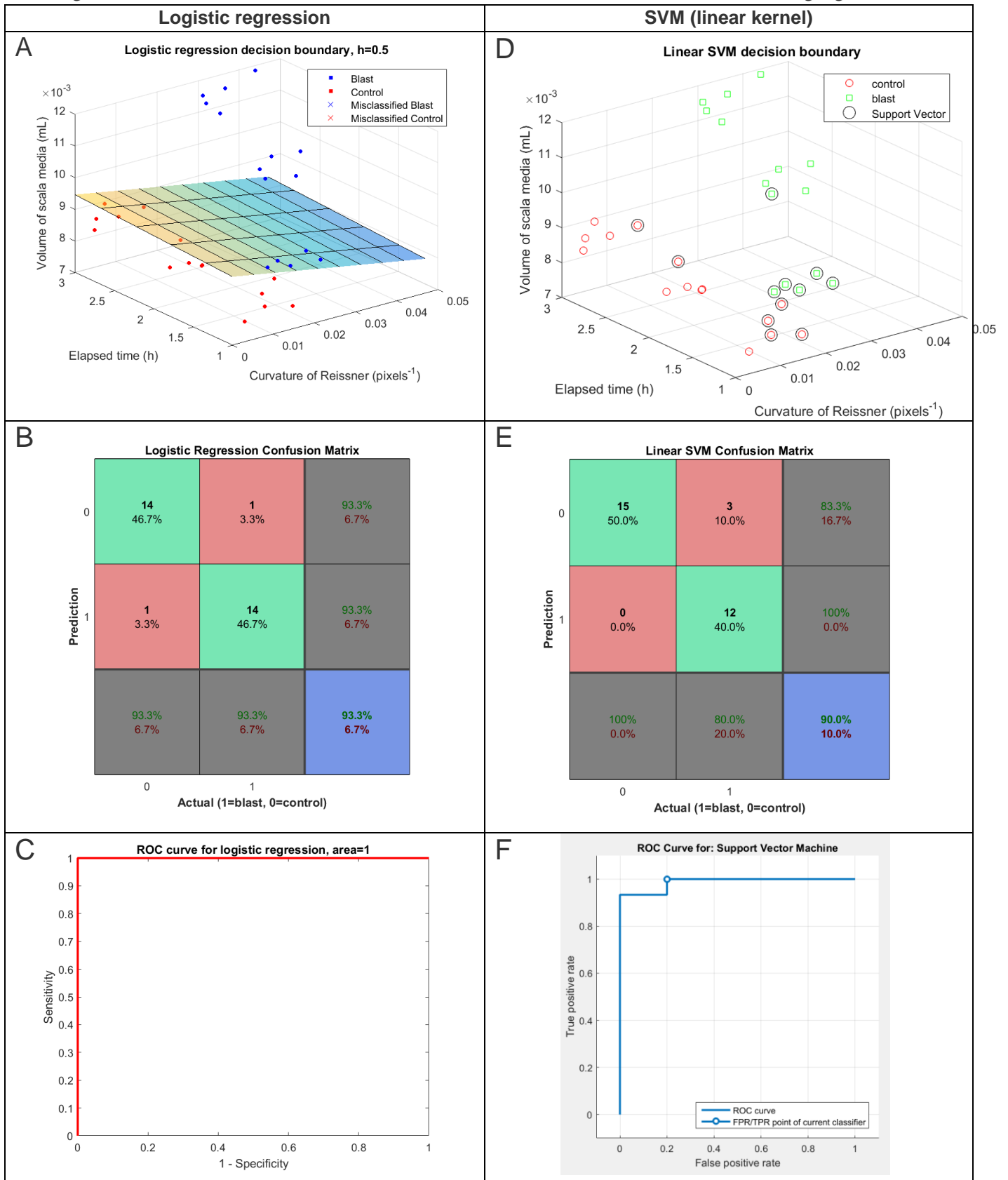
Logistic regression was performed on the training set using all the data (m=30 training examples, n=3 features) first to determine a training error. Using all data for training the model, the data is separable so no misclassifications are made. The model and its error-free classification of the training data is shown in figure 2A. During leave-one-out cross-validation, logistic regression yielded 1 false-positive and 1 false-negative (figure 2B). These errors were made for feature measurements at 1 hour of post-treatment where the data is less well separated (data not shown). The receiver operating characteristic (ROC) curve is shown in figure 2C. The curve was generated by calculating the specificity and sensitivity of the trained model above on the training data, with the cutoff logistic score adjusted at one hundred linearly spaced values of $h_\theta(x)$ between 0 and 1. There is a tiny crease in the top-left corner that is virtually negligible. It is surprising (and perhaps an error) that the area under the curve (AUC) is virtually one. More data is probably needed to create sizable and distinct training, cross-validation, and testing sets to flush out the ROC curve deflection for logistic regression.

Linear SVM produced the training model, LOOCV confusion matrix, and ROC curve shown in figures 2D-F. Similarly to logistic regression, cross-validation errors were made for feature measurements at 1 hour of post-treatment where the data is less well separated (data not shown). Naïve Bayes with Laplace smoothing yielded the same confusion matrix as the linear SVM algorithm (figure 2E). The LOOCV accuracies for logistic regression, SVM (linear, quadratic, and cubic kernels), and Naïve Bayes are summarized in Table 1. The differences in sensitivity and specificity among the algorithms are small.

The big takeaway from this initial exploration of machine learning classification of endolymphatic hydrops in mice OCT images is the better understanding of the problem at hand. Clearly, from the data plots (figures 2A, D), the features of cochlea in blast and control mice are well separated after at least 2 hours of exposure. Before 2 hours, the blast mice are too early in the development of endolymphatic hydrops from acoustic trauma to be as easily distinguished from the control mice. Therefore, in future work when more data is gathered, it will be essential to answer two questions: (1) Will data continue to be so well separated for acquisitions two hours or later after treatment? (2) Can we separate the 1 hour post exposure measurements better somehow? These results in mice provide insights into how these machine learning algorithms will translate to detecting mild and severe endolymphatic hydrops in human patients.

Table 1. Summary of learning models and their leave-one-out cross-validation accuracies.

Model	Leave-one-out cross validation accuracy (k=30)
Logistic regression (h=0.5 cutoff)	93.3%
Naïve Bayes	90%
SVM (linear kernel)	90%
SVM (quadratic kernel)	93.3%
SVM (cubic kernel)	96.7%

6. Conclusion/Future Work
Machine learning algorithms were able to accurately classify severe manifestations of endolymphatic hydrops. However it suffered when trying to separate milder hydrops at the one hour mark (data not shown). More data needs to be collected to confirm these results for a larger sample of endolymphatic hydrops from the population in blast mice and, if possible, in human patients with related hearing loss.

7. Acknowledgments

8. References/Bibliography

[1] Sloane PD, Coeytaux RR, Beck RS, Dallara J. Dizziness: state of the science. Ann Intern Med. 2001; 134: 823–832. pmid:11346317 doi: 10.7326/0003-4819-134-9_part_2-200105011-00005

[2] Centers for Disease Control and Prevention. Deaths: Final Data for 2013. NVSR Volume 64, Number 2. http://www.cdc.gov/nchs/data/nvsr/nvsr64/nvsr64_02.pdf

[3] Post R and Dickerson L. Dizziness: A Diagnostic Approach. American Family Physician. 2010 Aug 15;82(4):361-368. http://www.aafp.org/afp/2010/0815/p361.html

[4] Gates GA. Meniere's Disease Review 2005. J Am Acad Audiol 17:16–26 (2006). http://www.audiology.org/sites/default/files/journal/JAAA_17_01_02.pdf

[5] Committee on Hearing and Equilibrium. Ménière's Disease: criteria for diagnosis and evaluation of therapy for reporting. Otolaryngol Head Neck Surg 1995; 113:181–5.

[6] Lee HY, Raphael PD, Park J, Ellerbee AK, Applegate BE, Oghalai JS (2015) Noninvasive in vivo imaging reveals differences between tectorial membrane and basilar membrane traveling waves in the mouse cochlea. Proc Natl Acad Sci U S A 112: 3128–3133. doi: 10.1073/pnas.1500038112. pmid:25737536

[7] Cho S-I Il, Gao SS, Xia A, Wang R, Salles FT, Raphael PD, Abaya H, Wachtel J, Baek J, Jacobs D, Rasband MN, Oghalai JS. Mechanisms of hearing loss after blast injury to the ear. PLoS One [Internet]. 2013/07/11 ed. 2013 Jan [cited 2014 Aug 13];8(7):e67618. Available from: http://www.ncbi.nlm.nih.gov/pubmed/23840874 PMID: 23840874

[8] Jørgensen, Thomas Martini, et al. "Machine-learning classification of non-melanoma skin cancers from image features obtained by optical coherence tomography." *Skin Research and Technology* 14.3 (2008): 364-369.
[9] Lu, Hong. "Machine Learning Based Analysis of Coronary Stent Images in Intravascular Optical Coherence Tomography Pullbacks ." Electronic Thesis or Dissertation. Case Western Reserve University, 2015. *OhioLINK Electronic Theses and Dissertations Center*. 05 Dec 2015.

[10] Xu J, Ishikawa H, Wollstein G, Bilonick RA, Folio LS, Nadler Z, et al. (2013) Three-Dimensional Spectral-Domain Optical Coherence Tomography Data Analysis for Glaucoma Detection. PLoS ONE 8(2): e55476. doi:10.1371/journal.pone.0055476

[11] Kafieh R, Rabbani H, Kermani S. A Review of Algorithms for Segmentation of Optical Coherence Tomography from Retina. *Journal of Medical Signals and Sensors*. 2013;3(1):45-60.

[12] Burgansky-Eliash, Zvia et al. "Optical Coherence Tomography Machine Learning Classifiers for Glaucoma Detection: A Preliminary Study."*Investigative ophthalmology & visual science* 46.11 (2005): 4147–4152. *PMC*. Web. 6 Dec. 2015.

[13] http://www.intmath.com/applications-differentiation/8-radius-curvature.php

Code libraries:

- MATLAB Classification Learner application
  - Performed SVM with quadratic and cubic kernels with leave-one-out cross-validation
  - Obtained ROC curve for SVM linear kernel
  - Confirmed results of self-implemented code for SVM linear kernel
- MATLAB Statistics and Machine Learning Toolbox
  - Train and predict linear kernel SVM and Naïve-Bayes models
  - Display confusion matrices for logistic regression, SVM, and Naïve-Bayes