

Sep 19, 21 5:55	README.txt	Page 1/3
Amaan Rahman ECE 472: Deep Learning Professor Curro	Assignment 2 Binary Classification	
Remarks:	Attempts to implement ReLU, Leaky-ReLU, and Sigmoid activation functions were made and unsuccessful; the functions themselves have been fully and properly implemented, however an unnecessary amount of time was wasted on integrating the "handmade" activation functions into Tensorflow. The realization that "handmade" activation functions require integration was realizing that this very reason of no integration was causing my model to be unable to train due to failure of gradient computation. The quick solution that has been used instead was to utilize the built-in functions instead.	
	MultiPerceptron Design Considerations:	
Within my output layer).	One thing to note is that I don't include the input layer w discussion of design considerations (only hidden layers and	
My loss didn't converge. I then ramped the widths by 10 fold about and then I noticed convergence over 1500 iterations given a batch size of 32. This "funnel" design yielded losses to as low as 0.003 or possibly even lower. I tested out my final design, which is the "hourglass" configuration:	Initially, I decided on testing 8->4->2->1 setup, however my loss didn't converge. I ramped the widths up by about times 4, and it didn't converge. I then ramped the widths by 10 fold about and then I noticed convergence over 1500 iterations given a batch size of 32. This "funnel" design yielded losses to as low as 0.003 or possibly even lower. I tested out my final design, which is the "hourglass" configuration:	
	100->75->50->25->50->75->100->1	
	This design yielded optimal convergence compared to all permutations I have tested out thus far, yielding losses as low as 0.000002.	
	Citations:	
	Training function reference from Professor Curro's example	
	@misc{brownlee_plot_2020, title = {Plot a {Decision} {Surface} for {Machine} {Learning} {Algorithms} in {Python}}, url = {https://machinelearningmastery.com/plot-a-decision-surface-for-machine-learning/}, abstract = {Classification algorithms learn how to assign class labels to examples, although their decisions can appear opaque. A popula	

Sep 19, 21 5:55	README.txt	Page 2/3
r diagnostic for [âM-^@{}], language = {en-US}, urldate = {2021-09-19}, journal = {Machine Learning Mastery}, author = {Brownlee, Jason}, month = aug, year = {2020}, file = {Snapshot:C:\\Users\\Amaan\\Zotero\\storage\\XRT BM7XD\\plot-a-decision-surface-for-machine-learning.html:text/html}, }		
	@misc{noauthor_archimedean_2021, title = {Archimedean spiral}, copyright = {Creative Commons Attribution-ShareAlike License}, url = {https://en.wikipedia.org/w/index.php?title=Archimedean_spiral&oldid=1039754847}, abstract = {The Archimedean spiral (also known as the arithmetic spiral) is a spiral named after the 3rd-century BC Greek mathematician Archimedes. It is the locus corresponding to the locations over time of a point moving away from a fixed point with a constant speed along a line that rotates with constant angular velocity. Equivalently, in polar coordinates (r, î) it can be described by the equation $r = a + b \cdot \hat{i}$ with real numbers a and b. Changing the parameter a moves the centerpoint of the spiral outward from the origin (positive a toward î, = 0 and negative a toward î, = îM-^@) essentially through a rotation of the spiral, while b controls the distance between loops. From the above equation, it can thus be stated: the position of particle from the point of start is proportional to the angle î, as time elapses. Archimedes described such a spiral in his book On Spirals. Conon of Samos was a friend of his and Pappus states that this spiral was discovered by Conon.}, language = {en}, urldate = {2021-09-19}, journal = {Wikipedia}, month = aug, year = {2021}, note = {Page Version ID: 1039754847}, file = {Snapshot:C:\\Users\\Amaan\\Zotero\\storage\\XXA LYR9E\\index.html:text/html}, }	
	@misc{noauthor_python_nodate, title = {python - {Pandas} \& {Matplotlib}: {Plot} a {Bar} {Graph} on {Existing} {Scatter} {Plot} or {Vice} {Versa}}, shorttitle = {python - {Pandas} \& {Matplotlib}},	

Sep 19, 21 5:55	README.txt	Page 3/3
<pre>url = {https://stackoverflow.com/questions/49991227/pandas-matplotlib-plot-a-bar-graph-on-existing-scatter-plot-or-vice-versa}, urldate = {2021-09-19}, journal = {Stack Overflow}, file = {Snapshot:C:\\Users\\Amaan\\Zotero\\storage\\HS537TGY\\pandas-matplotlib-plot-a-bar-graph-on-existing-scatter-plot-or-vice-versa.html:text/html}, }</pre>		