Udacity Data Analyst Nanodegree (DAND) Review

William Koehrsen

May 21, 2017

I may seem like a strange student for one of Udacity’s courses. I am currently enrolled as a double major in aerospace and mechanical engineering at Case Western Reserve University where I have one year remaining. I am not actively looking to switch to data analysis, but wanted to augment my skills and ensure that I at least have some experience in this field that will only grow in import as the amount of data collected in every field continues to increase.

**My Background**

Udacity states that this is an intermediate program and “entering students should have programming experience (preferably Python), and ideally some background in descriptive and inferential statistics.” That described me pretty well at the outset. I had no formal training in Python but had picked some up on my own and felt relatively comfortable with the basic programming structure and the syntax of Python. I also had never taken an explicit statistics course, but had used statistical methods for numerous classes. I began this program while on an internship at NASA as I way to spend my weekends and evenings and any slow moments at work (there were a lot of times I was able to “work” on this class while on the job). I would say that Udacity’s assessment of where you should be before starting this course is spot on, and at least a minimal level of programming and statistics would be recommended. It would be possible to finish this degree with no experience whatsoever, but it would take a lot of self-teaching and augmentation of the course with other resources (perhaps other Udacity courses). If you are self-motivated, then nothing in this program should be that great of a challenge, and even for those with extensive experience, there will be a lot of time spent exploring the Udacity forums or Stack Overflow.

**Time and Cost**

Before I even talk about the content of the course, I will address the two questions I inevitably receive whenever I bring up the nanodegree program: “How long does it take?”, and “What does it cost?” In answer to the first question, Udacity estimates that students typically take around 260 hours to complete the course. I am not sure where that figure comes from and whether it includes both time spent watching lessons and working on the projects, but it seems a reasonable figure. At the minimum advised 5 working hours/week, that would take 52 weeks, or exactly 1 year to complete the course. I spent a total of 315 hours on the nanodegree over the course of 2 months. Like I said, I mostly worked evenings and weekends, and averaged about 35 hours per week over the course of the degree. Of that time, 135 hours was spent on projects, and 180 hours were spent on the lessons/problem sets. Two months might be pushing it for most students, but it is manageable with a large commitment.

In terms of cost, my answer always starts out: “Well, the answer could be free.” All course materials and even project materials can be accessed for free! However, in order to have your projects reviewed and receive an official graduation certificate, you’ll have to pay. In my case, I choose the basic level of enrollment at $200 per month. There is also the option of the nanodegree plus, which runs $300/month but offers a job guarantee in the field within 6 months of completing the course or your money back (subject to [terms and conditions](https://www.udacity.com/legal/nanodegreeplus), including submitting at least 5 applications per week). Both levels of enrollment offer a 50% refund on tuition if the program is completed within 12 months. That means I paid a total of $200 for nearly 200 hours of instruction. At my university, an average semester-long class will involve about 100 hours of instruction and costs $1900/credit hour resulting in a cost of $57 per hour of instruction. I am not claiming that the Udacity lessons are the exact equivalent of a college course, and they are videos rather than in person teaching, but even so, at 1/50 the cost, I am more than pleased at the value of this degree.

**Overall Impression**

I will get into the details of each project further on, but for those looking for my final takeaway, I would say that I fully recommend this course. The quality of all aspects of the course surpassed any expectations I held, and after this experience, I will be enrolling in my next course as soon as possible. I have always been excited at the prospect of learning new skills, and the first day of a college course has long been my favorite moments when I look at the syllabus and see everything that I will be able to find out about. However, by the end of the semester, this initial eagerness has faded into the maelstrom of stress and frustration that arises around any exams. Moreover, I often finish a course only to realize that I know nothing about how to apply anything I’ve just spent the past four months learning. Sure, I can recite the relevant facts, and perform adequately on the tests specifically designed to align with the material, but I have no idea how I could take what I learned and do something useful with it. Udacity has figured out how to solve this problem, and that solution, not surprisingly, is projects. Every section of the Udacity course is followed up by an in-depth project that usually involved real-world data (or allowed students to find their own dataset) and required me to apply nearly everything I had learned in the instructional videos. These projects are no simple fill-in-the-blanks exercises either. The level of involvement varies, and some of them (including the machine learning project) are fairly structured, but even so, countless times I found myself digging through Stack Overflow posts and skipping meals to work on the projects because they were so engaging and I wanted to do my absolute best. I remained almost 100% engaged throughout the entire course (there were a few slower spots at the beginning of some of the lessons) and did not find myself with the usual subject fatigue that I often develop at the end of college courses where I just want to be done and never have to look at the material again. Even after finishing a lesson, I would find myself reading blog posts or digging into some application of the material I had found online. Rather than make me sick at the subject, the Udacity lessons made me genuinely curious about the topics, which feeds back into a virtuous learning cycle where I spend nearly all of my free time exploring more and more of the world that this course opens up into data analysis.

Tools in my repertoire gained from Udacity: Version Control (Git/GitHub), Jupyter Notebooks, Markdown, Python, d3.js, Javascript, HTML, CSS, R programming language, Tableau, SQL

For me, the entire strength of the Udacity DAND program is the application based philosophy of the course. If you are looking for the theoretical underpinnings of data analysis and visualization, then perhaps this is not the right course for you. If however, you want to get your hands dirty and work on projects that reflect what data analysts actually do in the real world, then sign up as soon as possible. All of Udacity’s courses are developed with industry partners, and this course is a partnership between Udacity, Facebook, and Tableau (a maker of a [great data visualization tool](https://www.tableau.com/products/desktop) that gives you no reason to ever make an Excel chart again). Therefore, the focus of the class is on developing skills that will actually be put to use as opposed to learning about the principles of the field. There is conceptual material ([Edward Tufte's](https://en.wikipedia.org/wiki/Edward_Tufte) work and ideas make several appearances) that makes its way into the course, but it is always reinforce with a real-world application and usually several quiz questions. I have often struggled with the question of whether or not it is important to understand the theory of a concept or if it is enough to know how to use the tools in a field. Udacity certainly falls on the tool-usage side of that line and I thoroughly enjoyed this approach. There was no talk of abstract concepts that often lose students (although there were numerous links for papers and articles for those wishing to dig into the theory) but merely the presentation and explanation of the actual tools in use today and how they are best used.

The structure of the Udacity DAND is straightforward. There are seven required projects each with an associated series of instructional videos. The videos are broken down into several lessons. The format varies slightly across projects, but the basic form is a video, followed by a simple programming exercise or quiz question. Some of the lessons are followed by slightly more involved problem sets, and once all of the lessons for a project have been finished, it’s time to start working on the project. Like I said, most of these are fairly involved and the time commitment for the project can vary widely from about 2 hours for the first introductory project to over 30 for the data visualization project. The projects were far and away some of the most enjoyable time I have spent on education. In addition to the core curriculum projects, there are also numerous “extracurricular resources.” These include additional projects that are optional (but highly recommended), how to use tools such as Git and GitHub, and career development tools (resume reviews, interview practice, etc.). I did not get as involved with the career development aspects as I am not actively looking for a job in the data analysis field, but I did take advantage of several of the extracurricular options and was pleased. Overall, the amount of instruction and resources available surpassed many of my college courses, and for the price of a single textbook, the Udacity Data Analyst Nanodegree Program was an exceptional learning experience.

**Project Reflections**

For those who only wanted the basic overview, feel free to leave now before I get lost in the details! I thought that I would give a brief run-down of the eight (the curriculum has now switched to seven) projects that I completed over the course of the degree. First, a quick note about the update of the curriculum. I was working on the final required project, A/B testing, when Udacity officially changed the entire curriculum of the course. The A/B testing project was made optional, and the data visualization course was switched from being based on d3.js to Tableau. This review mostly focuses on my experience, but the majority will be relevant for the new curriculum as well. I did go on to finish the A/B testing, because once I start something I cannot let it go for better or worse, and I also went through the lessons for the data visualization using Tableau project (although I did not complete the project as I already had finished the data visualization project using d3.js). I believe that these changes are for the best, as the d3.js project was exceedingly involved (although I certainly enjoyed getting lost in the coding!) and the A/B testing project was the low point of the course. I assume that is what Udacity found as well and I applaud them for their efforts to keep the material as relevant as possible. I provide the times spent on introductory material which includes watching lessons and completing the small quizzes/ problem sets, as well as the time spent on the project for each project. I tracked all my time in [Toggl](https://www.toggl.com/) which I must recommend for anyone interested in maximizing their efficiency. Being aware of how much time is actually spent on different tasks is invaluable and taught me a lot about my habits. At the very end, I also include a section discussing the resources available, which I found extremely impressive.

**P0: Nanodegree Orientation; 0.5 hours on Introductory Material, 2.25 hours on project**

[**Final Version of Project**](https://github.com/WillKoehrsen/Udacity-Data-Analyst-Nanodegree/blob/master/Project%20Final%20Versions/P0%20Final/Bay_Area_Bike_Share_Analysis.pdf)

There is not much to say about the first project other than that it was a decent introduction to the style of the Udacity course. The most useful aspect of this project was the introduction of Jupyter Notebooks (formerly IPython Notebooks). Jupyter Notebooks allow you to write code and reports using Markdown in the same document in the browser. I went on to use Notebooks in three other projects, and my initial reaction was amazement at this tool. I would come to find that the perhaps the real value of the Nanodegree is exactly that; a brief exposure to all the available tools. It is impossible to predict which tool you will use every day in a job, and being exposed to as many as possible is a great approach to be prepared for a career in data analysis. The project was relatively straightforward and involved walking through a prepared notebook, creating a few simple graphs, and answering several questions using the data. This served as a gentle introduction to the degree and gives you a first chance to see the value of a code review.

**P1: Statistics**; 10 hours introductory material, 3 hours on project

[Final Version of Project](https://github.com/WillKoehrsen/Udacity-Data-Analyst-Nanodegree/blob/master/Project%20Final%20Versions/P1%20Final/P1%20Stroop%20Effect%20Analysis.pdf)

The lessons for this project start off extremely basic and are definitely skippable for those with even a rudimentary knowledge of statistics. I have a completionist attitude though and found myself watching every single video and do every single quiz even when that involved having to listen to the description of the mean, median, and mode. Nonetheless, I always have found value in reviewing even the most basic material, and the lessons quickly moved into more advanced hypothesis testing concepts such as t- and z-tests, and I had to pay close attention in order to complete the quizzes. Here was my first exposure to the Udacity style of teaching videos which mostly involve a narrator talking while concepts are written out on a white background in a sort of Khan Academy-esque manner. I find this style to be more effective than being lectured at and I am glad that Udacity mostly sticks aware from lecture type videos and keeps all the videos short (the longest video in the entire course was about 8 minutes). There are several videos that are just people talking to the camera, but these are mostly framed as discussions and do not have the professor-to-students feel of a typical college lecture. This project gave me a very valuable experience, my first failed review! I believe that traditional education has become obsessed with reducing student mistakes and encouraging perfection to the detriment of actual learning. By not assigning grades and operating on a fail/pass system with unlimited submissions, Udacity has solved that problem. As in the real world, nothing comes out perfect the first time, and most of my learning occurred when I was trying to fix mistakes I had made on an initial submission. Moreover, every single review I received, even those in which I had passed the project, were thorough and clearly my projects had been examined in some detail. This project was all based in Excel (or the spreadsheet application of your choice) and was not a serious time commitment even with a revision or two. The report was simply a Word document that involved answering a set of questions provided by Udacity. I enjoyed this project despite the slow start and was encourage by the feedback I received.

**P2: Intro to Data Analysis; 17 hours on introductory material, 16.5 hours on project**

[Final Version of Project](https://github.com/WillKoehrsen/Udacity-Data-Analyst-Nanodegree/blob/master/Project%20Final%20Versions/P2%20Final/Data_Analysis%20_with_Python.pdf)

Project 2 (the order of projects has changed in the new curriculum but the project subject matter remains the same), was where the programming aspect of the course begins. The lessons combine material on the data analysis process with programming instruction all in Python. The Python programming language is without a doubt the best option for this course overall. The ability to write useful programs within a few hours of first learning the language is invaluable to the confidence of beginners, and the number of modules and libraries for working with data mean that nearly any source of data can be cleaned, analyzed, and visualized with relative ease without ever leaving Python. The primary focus of the course was using NumPy (numerical python) and Pandas dataframes (a spreadsheet-like data object) for processing and then visualizing data. The lessons provide most of the background needed to get started on the project, but beginners to Python might need to dive into some other courses to build up a basic level of familiarity with the language. I already had some experience with Python but found the course to be a useful refresher and the further elaboration on the capabilities of the Jupyter Notebook was invaluable for completing the project. The project itself was pretty easy to get started on because Udacity provides you with two datasets that you can choose from to analyze. At this point in the course, the focus is not on data cleaning/wrangling (that will show up shortly enough!) and giving students two clean datasets to work with I can imagine helps to reduce frustration and drop out. It is clear that Udacity has thought out every detail of the course (or have gathered enough data to make well-informed decisions). In contrast to professors who often seem to delight in seeing students struggle and brag about the difficulty of their exams, Udacity makes every attempt to facilitate applicable learning and build confidence in their students. For some, this means that the initial projects will not be a great challenge, but I think that is worth it for the eventual success of the majority of the students. I ended up using a Jupyter Notebook for both the coding and the report itself (written using Markdown, another tool that I have in my repertoire thanks to Udacity) and again, I failed the project on first submission (Udacity would never use the word failed, they prefer to say that the submission has some specifications that were not met). Again, the review forced me to think about my project and analyze the choices I had made in order to improve my submission. Even for the aspects of the project I had passed, I found myself making changes in order to increase the efficiency of my code purely out of a desire to end up with the best project possible. The goal of the project was to investigate either dataset (the famous [Titanic passenger data](https://d17h27t6h515a5.cloudfront.net/topher/2016/September/57e9a84c_titanic-data/titanic-data.csv) or an incredibly detailed [corpus of baseball statistics](http://www.seanlahman.com/baseball-archive/statistics/)) and come up with several questions to guide the inquiry. I choose the baseball dataset and ended up getting way too involved with the data. I was not even interested in baseball on the surface, but I wanted to see what the data could tell me. I probably spent more time than was required to complete this project and even tried some rudimentary machine learning on the data out of curiosity. I found this trend of going much farther than was required repeating itself over and over again in the DAND. The enthusiastic comments from the reviewers as well as the open-ended nature of the projects encouraged to continue expanding and expanding until I had done as much as was reasonably possible.

**P3: Data Wrangling; 29 hours on introductory material, 24.25 hours on project**

[**Final Version of Project**](https://github.com/WillKoehrsen/Udacity-Data-Analyst-Nanodegree/blob/master/Project%20Final%20Versions/P3%20Final/Data_Wrangling_with_Python.pdf)

While the data for the previous project was provided cleaned and ready to go, the entire point of this project was to learn how to clean a dataset. The dataset in question was any region of the [OpenStreetMap](https://www.openstreetmap.org/#map=5/51.500/-0.100), a user-created and maintained map of the world that features plenty of opportunities for cleaning. The lessons were focused on the principles of data cleaning and using a SQL ([structured query language](https://en.wikipedia.org/wiki/SQL)) database to organize data for later analysis. This project was the first that felt to me like a complete package. Not only do students have to download real-world data and clean it, but they also have to insert it into a SQL database and use Python to query the database in order to extract usable information. Whether or not this is something that actually occurs in industry is beyond my knowledge, but it felt like quite an achievement to work through all the steps. Again, Python was used extensively, for the cleaning of the raw data (which was downloaded as XML), the insertion into a CSV file, and then querying the SQL database (using SQLite3) to figure out any interesting aspects of the area of the map being studied. The basic process was download the raw data as XML, clean the data using Python, move the data into a comma-separated file, create a SQL database from the CSV files, query the SQL database using Python, and record any information learned about the data. This project was certainly frustrating at times because of the sheer number of new concepts covered but it was more of the frustration that makes me want to stay up all night working on the project rather than the frustration that makes me want to quit and get rid of my computer. Again, I was impressed with both the lessons and the project. The best aspect of these open-ended projects are that they can be as challenging as you want. If you want to do the absolute minimum and pass the project so you can get that sweet piece of paper, that is perfectly fine, but I think that if other students had my experience, that was my outlook at the beginning, but I soon found myself doing as much as I could on each project out of pride. While I occasionally am proud of a high test mark I receive in college, there is nothing that compares to making something that I would actually want to show other people (or potential employers). Moreover, I was driven out of a genuine desire to find out more about the data. I choose to analyze Cleveland because I thought that maybe I could learn a few things about the city, and amazingly, out of all those thousands of lines of XML, I did learn a couple things (I won’t get into detail here as this review is already unmanageable, but take a look at the final version if you are interested). Again, I think the tools that you are exposed to in this course are more valuable than the actual concepts, and the number of skills I acquired during this one course alone was astounding. Over this course, I felt like I was receiving only the most pertinent information, and Udacity was essentially filtering out all the unnecessary information that I would have received in a typical college degree in data analysis.

**P4: Exploratory Data Analysis; 30 hours on introductory material, 23 hours on project**

[**Final Version of Project**](https://github.com/WillKoehrsen/Udacity-Data-Analyst-Nanodegree/blob/master/Project%20Final%20Versions/P4%20Final/Exploratory_Data_Analysis_in_R.pdf)

The focus of the 5th project in the DAND is exploratory data analysis, or taking a dataset and seeing what it can tell you with no preformed hypotheses. The lessons work their way up in complexity, from analyzing a single variable to multivariate analysis. This project is based completely in R, which is might be a sticking point for some. I found R to be have a steep learning curve and the syntax was completely different from any other language I have worked with (mostly MATLAB, Python, and Javascript). However, after plenty of time spent on various forums and learning as much as possible, I found the abilities of R to be incredible in terms of data visualization. I found the lessons a little too shallow in their coverage of R, but there are so many other resources of the web and R is such a complex language that to try and teach all the specifics is probably not the best use of student’s time. The project itself involves exploring a dataset, with several cleaned options provided by Udacity as well as the choice to select your own dataset. I choose to use my own dataset [from Kaggle’s collection](https://www.kaggle.com/datasets) and got a little more practice in cleaning data. The project was very unstructured, and while some students may find this freedom intimidating, it also allows for a wide variety of approaches. It seems that with every new programming language, there is a period where, no matter how hard I concentrate, I cannot understand the language and I struggle to do anything useful at all. This usually continues for a number of hours, until a certain point at which everything starts to click and I slowly gain confidence until I reach a point where I feel like I can write several lines of code without having to turn to Stack Overflow. Once that happened with R, I started to enjoy the project and ended up writing quite a bit more than I had anticipated. I would have preferred a more thorough introduction to R in the lessons, but again, Udacity cannot teach all the intricacies of every available tool, and just as in college courses, almost all of the learning will happen on the problem sets and projects anyway. This ended up being my second favorite project, and I think it was in small part due to the amount of learning I had to do over the course of the project. My recommendation would be to explore your own dataset and find something that genuinely interests you as it makes the work more rewarding. I do not know how widespread the use of R is in industry, but I am glad to have learned how to use it and I created another work that I am actually proud to show others.

**P5: Intro to Machine Learning**; 32.15 hours on introductory material, 24.25 hours on project

[Final Version of Project](https://github.com/WillKoehrsen/Udacity-Data-Analyst-Nanodegree/blob/master/Project%20Final%20Versions/P5%20Final/Enron_Machine_Learning_with_Python.pdf)

Upon looking at the syllabus for the course, this was the project that most interested me. I have heard much about machine learning and worked through some of [Sentdex’s tutorials](https://www.youtube.com/user/sentdex) on YouTube (I highly recommend all the videos on this channel and Sentdex’s website [Python Programming](https://pythonprogramming.net/)) but I did not have a real idea of machine learning and how it was employed. This was a basic introduction to the concept and the application but I found it to be more than enough to get me interested and I think that my next Udacity course will be the [machine learning engineer nanodegree](https://www.udacity.com/course/machine-learning-engineer-nanodegree--nd009). It would be impossible to teach all the concepts of machine learning in a single class, and Udacity once again goes easy on the concepts and focuses on the applications. All of the coding is done in Python using the wonderful SciKit-Learn library that is great for getting a classifier up and running a few lines of code. The lessons cover most of the basic machine learning algorithms including decision trees, support vector machines, and clustering as well as briefly introducing some more complex topics such as principal component analysis and text learning. The project is fairly structured and follows a typical path with the basic structure of most of the programs provided already by Udacity. The main parts of the project are selecting the features to use (or making new features), selecting a classifier, tuning the parameters of the classifier on the dataset, and validating the classifier. All of the work is done on the Enron email and financial dataset. This has become a fairly standard dataset for use in machine learning examples and does not require much cleaning before it can be used. Again, Udacity focuses on getting students programming and applying concepts rather than worrying about all the peripherals and details that mostly frustrate students (but will have to be covered eventually). Even with the structured format of the project, I struggled initially as there are a lot of new concepts that are can only be briefly covered in the lessons. Nonetheless, after enough work and time spent looking through forums, I was able to complete a decent machine learning project and it was enough to convince me that I want to pursue further education in this field. I think machine learning will be critical to anyone doing data analysis, and I applaud Udacity for including a brief introduction in this course. It may not be enough to get a student into the field, but it provides a nice starting pointing and a look through the hype that surrounds machine learning and actually shows how a typical project would be completed.

**P6: Data Visualization; 39.75 hours on introductory material, 33 hours on project**

[**Final Version of Project**](https://github.com/WillKoehrsen/Udacity-Data-Analyst-Nanodegree/tree/master/Project%20Final%20Versions/P6%20Final) **(this will need to be downloaded and run on a local server to see the visualization)**

The data visualization course has been switched from d3.js to Tableau, so this may be a little outdated. Evidentially, Udacity decided that d3.js was a little too frustrating to learn and the Tableau would be a better choice because it is used more and features less time to get an interactive visualization up and running for beginners. This was my favorite project, and I am sad to see that d3 will no longer be required, but it is probably for the best. Much as with the R project, Udacity attempted to teach an entire new language in one project, and they might have overreached. I loved digging through forums and documentation to try and figure out why my code was not working, or how to position an element just right, but that is not for everyone. Tableau is a GUI-based visualization tool that requires no programming and therefore will be much less frustrating and easier to use. The course itself was a little monotonous at times, and I do not think that it adequately prepared me for the project. However, there would be no way to teach everything that I had to learn for the project in the videos without making them hundreds of hours long. In the end, I believe that the switch to Tableau is a good decision that will improve the student experience. I have worked through the Tableau lessons, but have not gone through with the project as I completed it in d3.js (a visualization library built on top of Javascript). The learning curve for d3 was quite steep especially for someone with no prior experience in Javascript. I admit that I love the simplicity of Tableau, and while it is at a high level of abstraction, it still features an impressive level of flexibility and allows for many data operations that cannot be found in a spreadsheet software such as Excel. I do not want to become too invested into any one technology, and I think that the switch to Tableau for this project is the right choice. I also think that Tableau must have wanted to sponsor the course which is fair enough for Udacity because the respectability aspect of having another company attached to the degree is worth the change in curriculum. Although I am proud of my visualization using d3, I recognize that technologies change, and I must be willing to adapt with the rest of the world. If d3 is a technology being phased out, and there are better tools available, I see no reason for Udacity students to continue to struggle through this project just because that is the way it was in my experience. I learned a ton, but it required a lot of investment, and I see no reason to continue to cling to an outdated tool when clearly better options are now in practice.

**P7: A/B Testing; 10.25 hours on introductory material, 8.25 hours on project**

[**Final Version of Project**](https://github.com/WillKoehrsen/Udacity-Data-Analyst-Nanodegree/blob/master/Project%20Final%20Versions/P7%20Final/AB%20Testing.pdf)

The other major change that Udacity made to the curriculum was to switch the A/B Testing project from required to optional. Again, I have to say that I agree with Udacity on this decision. This project was the low point of the entire degree, which is to say that I still enjoyed it more than a typical college course. The project was again based entirely in spreadsheets, which was a little disappointing after I had been so absorbed in the code in the data visualization project. My main complaint with this project was that it felt extraneous. It came at the very end of the degree, and too many of the videos were discussions rather than the narration over writing and working out problems that formed the basis of most of the other lessons. I found it difficult to stay focused on these lessons because it seemed like there was little application of the material that was discussed. I appreciate that Udacity included this lesson, and that it is still available as an “extracurricular” project for those who want to augment their skill set, but it seemed like a project that was tacked on because the content was simply there. The project itself was not that difficult, and again, I received an excellent review of my first submission that I was able to iterate on for a second, successful submission. I didn’t need to complete this project because it was made optional while I was still enrolled, but because I had already started, I was determined to finish it and I am glad that I did. I cannot recommend that future students in the program complete the A/B testing project though as I am sure that there are better resources out there for those who are interested in the subject matter. The A/B Testing project was a good idea, but the implementation was not up to the standards I had come to expect from the other projects in the course.

**Resources/ Thoughts on Curriculum Update**

Besides all of the projects and lessons, Udacity provides many other resources for those committed to becoming a data analyst. Most of the extracurriculars are focused on career development, and I did not have a chance to explore many in-depth besides the resume and cover letter review (the Udacity reviewer was kind enough to provide feedback on a cover letter I wrote for an aerospace engineering job which I appreciated). I have little doubt that for those intent on becoming a data analyst, this course by itself could be sufficient to secure a job in the field. If I wanted to make the switch, I probably would not have paid for the “plus” version of the course, but I would have taken advantage of all the additional resources and made a serious attempt to develop a curated, professional portfolio and even gone through the interview preparation. Despite my non-completion of all extracurriculars, I was extremely satisfied with the value of the course. I got everything I wanted from this nanodegree and more. There were certainly struggles at times, but as in the real world, not everything will be explained to you at the outset and the most valuable skills one can learn are how to find answers that already exist, and how to implement and improve existing solutions. The Udacity class forums are a rich place of information, and there are several instructors that are very active on the forums. I personally found that I never needed to post my own question because anything I could think to ask had already been addressed. For those willing to commit a large amount of time and concentration to the course, it could really serve as the foundation for a career change. The forums, combined with all of the other resources form a complete package that pretty much replicates what one would find around a college course with opportunities for interacting with other students and even instructors. Moreover, with the curriculum update, it is clear that Udacity is committed to ensuring that this course remains relevant for data analysts and is continually evaluating the course to make sure that it reflects the current state of the field. Part of the reason I choose Udacity over other options such as Stanford and Coursera was the institutional backing of the various companies that sponsor each nanodegree. With both Facebook and Tableau behind this course, I think that it will remain relevant well into the future even with the rapid pace of change that characterizes any technological field. Udacity has managed to create a great course and business model that occupies an ideal spot between free online MOOCS and expensive college courses that do not reflect the state of the working world. The Udacity Data Analyst Nanodegree receives my highest recommendation and was a top-notch learning experience.