Master of Science in Data Science Readiness Assessment

What is this assessment and why should I complete it?

We want to ensure you have all the tools you need to succeed in the online Master of Science in Data Science program. That's why we have created the optional readiness assessment, designed to help you gauge your current skills and experience in data science. This assessment is entirely for your benefit-it will not impact your admission to the program. Instead, it serves as a helpful way for you to understand how well your existing knowledge aligns with the background knowledge needed to be successful in this program so you may fully achieve the program learning outcomes listed below. Consider this a starting point on your journey to mastering data science.

We recommend you complete this assessment before applying for the program. You may use any programming or calculation tools you want to complete the assessment, and the answers are provided.

If you have any additional questions about this readiness assessment or the online Master of Science in Data Science program, you can contact: poapply@purdue.edu.

Program Learning Outcomes:

- **Programming & Technology Integration** – Learners will develop expertise in programming languages, gaining the ability to design and implement data-driven solutions. Learners will apply advanced technologies, including cloud computing and big data frameworks, to effectively handle and process large-scale datasets.

- **Machine Learning Applications** – Learners will showcase a deep understanding of ML algorithms and models, applying them to real-world scenarios. Learners will demonstrate the ability to develop, implement, and optimize ML solutions for tasks such as classification, regression, clustering, neural networks, and recommendation systems.

- **Applied Data Analysis** – Learners will demonstrate proficiency in collecting, cleaning, and analyzing diverse data sets using advanced statistical and ML techniques. Learners will apply data analysis skills to derive actionable insights, make informed decisions, and solve complex problems across various domains.

- **Applied Data Visualization and Communication** – Learners will perfect the craft of data visualization and communication, creating compelling visual representations of complex data to effectively convey insights. Learners will apply storytelling techniques to communicate findings clearly to both technical and non-technical stakeholders.

- **Ethical and Responsible Data Science Practices** – Learners will adhere to ethical standards in data science, demonstrating a commitment to privacy, transparency, and fairness. Learners will apply ethical considerations in all stages of the data science lifecycle, ensuring responsible use of data and technologies in addressing societal challenges.

READINESS ASSESSMENT FOR MASTER SCIENCE IN DATA SCIENCE

Programming:

Overview: Do you have any experience with programming? Do you have any experience with Python or any other high-level scientific programming language like R or Matlab?

1) Basic programming:

a) What is a 'for loop' and when would you use it? What about a 'while loop'?
b) What are functions, and why are they useful?
c) What is the difference between an integer and a float?
d) Can you create a scatter plot in Python or R?

2) Intermediate programming:

a) Can you write code (in any programming language) to count the number of zeros in an object/vector/array called dataset?
b) What is a pointer/reference and why is it useful?
c) Given a dataset of 1-dimensional observations, write code to standardize it (i.e. subtract out the mean and then divide by the standard deviation)
d) Have you used command line tools and automated programming interfaces (APIs) in any setting? This is important because we will be using these during the course of the program to upload and analyze data on cloud-based computing platforms like AWS and Google Cloud.

3) Math and probability:

a) If A is an m-by-n matrix and B is an n-by-p matrix, what are the dimensions of AB? What is the formula for element (i,j) of this matrix?
b) If you roll 2 fair 6-sided dice what is the probability at least 1 of them returns a 3? What is this probability if you roll 100 dice?
c) What is the Gaussian/normal distribution and why is it useful?
d) If X is a random variable with probability p(X), what is the formula for the mean of a X?
e) What is the derivative of a function, and why is it useful?
f) What is a linear regression?

Answers:

1a: In programming, a 'for loop' allows allows you to control the flow of a program, repeatedly executing a block of code a fixed number of times. E.g. you might have a dataset of people and want to execute the same set of commands on each person's data. A 'while loop' repeatedly executes a block of code while some condition holds true. E.g. instead of processing each person's data in the previous example, you process people's data until you have found 50 people who meet some admissions criterion.

1b. A function is a named block of code that can be called to apply a set of computations to any input that meets its specifications. Functions allow you to avoid repeating the same lines of code in different parts of the same program/across programs, allowing you to break your overall program into simpler modules. This makes is easier to debug, maintain and share your programs.

1c. A float or floating point is how a computer represents and stores a number that is not necessarily an integer (e.g. 0.5, pi). Floating point numbers are more flexible than integers, but take more memory and are slower to do computations with.

2a. Pseudocode:

```
count = 0
for(datapt in dataset)
  if(datapt=0) count = count+1
```

Or

```
count = sum(dataset == 0)
```

2b. A pointer references the location of an object (e.g. a dataset) in the computer's memory, rather than containing the object itself. Among other things, points make it unnecessary to copy/duplicate big datasets so that different parts of the program can access it.

2c.

Approach 1: Pseudcode for brute force approach:

```
%%%%%%%%%%%%%%%%%%%%
      mn = 0
      sd  = 0

     % Calculate mean and standard deviation of dataset

      for(datapt in dataset) {
        mn = mn + datapt
        sd  = sd    + datapt*datapt
     }
     N = len(dataset)
     mn = mn/N
     sd  = sd/N
     sd  = sqrt(sd – mn*mn)  % Variance is mean of the square – square of the mean

    % standardize dataset
    for(datapt in dataset)  atapt = (datapt-mn)/sd

%%%%%%%%%%%%%%%%%%%%
```

OR Approach 2 using numpy

```
mn = dataset.mean()
sd  = dataset.std()
dataset = (dataset – mn)/sd
%%%%%%%%%%%%%%%%%%%%
```

3a. The result is an m-by-p matrix, with the formula given below (from Wikipedia, https://en.wikipedia.org/wiki/Matrix_multiplication)

If $\mathbf{A}$ is an $m \times n$ matrix and $\mathbf{B}$ is an $n \times p$ matrix,

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{pmatrix}$$

the *matrix product* $\mathbf{C} = \mathbf{AB}$ (denoted without multiplication signs or dots) is defined to be the $m \times p$ matrix

$$\mathbf{C} = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{pmatrix}$$

such that

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{in}b_{nj} = \sum_{k=1}^{n} a_{ik}b_{kj},$$

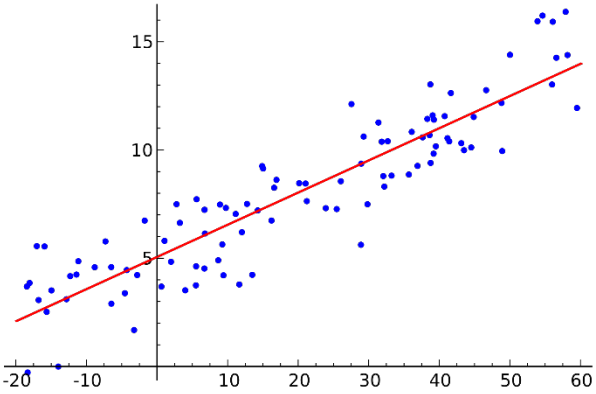
3b. The probability that 1 does not return 3 is 5/6. The probability that both do not return 3 is 5/6*5/6 = $\left(\frac{5}{6}\right)^2$, since the two events are independent. Thus the probability at least one of them returns a 3 is $1 - \left(\frac{5}{6}\right)^2$. For a 100 dice, this becomes $1 - \left(\frac{5}{6}\right)^{100}$

3c. The Gaussian/normal distribution, also called the bell curve, assigns probability to different possible outcomes of a real-valued continuous random variable. It is useful because of the so-called Central limit theorem, which states informally that if we add up a large number of random variables, then the distribution of the sum can be approximated well by the Gaussian distribution.

3d. If X is discrete, then the formula is $\sum Xp(X)$, while if it is continuous, it is $\int Xp(X)dX$. Here, the sum or integration are over the range of values X can take.

3e. The derivative of function at any input value gives a direction and rate of fastest change at that input point. If the function takes a scalar as input, then this is just the slope of the function any point.

3f. Linear regression characterizes the relationship an input and an output variable by fitting a straight line/plane through a dataset of measurements of these variables.



By Sewaqu - Own work, Public Domain, https://commons.wikimedia.org/w/index.php?curid=11967659