On The Distribution of Mile-Run Times at The University of Chicago Laboratory Schools

Ben Glick

May 3, 2016

AP Stats, Ms. Maguire, Inst.

Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.

Introduction:
I investigated the distribution of one-mile run times in PE classes at U-High. The bulk of my work was comparing samples against each other, according to various stratifying "filters" I applied on my data, but I also tested sample distributions against accepted national averages, to determine whether various groups at Lab run faster, typically than other students. My population of interest is therefore every student in gym class, which is every freshman, sophomore and junior, as well as a few seniors. I also compared data from the last two years against each other, meaning I was able to test if this year's high schoolers ran faster, typically, than last year's. The primary test statistic used was the T test statistic, found using one and two sample T tests.

Procedure:
In order to carry out this study, I accessed data that the PE department has on hand. Because every gym student runs the mile at least once, I could theoretically do a census of all gym students. However, due to the number of students there are in PE, I decided to do a random sample. I used python to carry out this sample, and important selections of code, as well as walkthroughs of said code, are attached at the end of this section. Because there are some students who run the mile more than once, I always took the faster of two times in the event I select a student that has run the mile more than once. Once I had the raw data, I examined it, and noticed a number of inconsistencies which would cause logistical issues. Many of those are discussed in the "*The Data*" section, below. These inconsistencies needed to be taken into account while I was sampling. An example of how this has to happen could be: a student did not run the mile, and thus, in the spreadsheet, they could be recorded as a number of things, none of which are acceptable data. My code had to recognize when that was happening and have a contingency plan. In this case, I simply disregarded this data point and randomly sampled another. Because stratifying reduces sample variability, I stratified my sample before creating four separate distributions, which I combined in various ways. I first stratified by grade, represented by graduation year, and then drew two samples of size 30 from each year's data, one for males and one for females. Every year, a few seniors take PE, and therefore have to run a mile. However, due to the small sample size, seniors were removed from this sample. The result of this is four acceptably randomized and stratified samples, which are as follows: (n=30 Males who ran in Fall of 2014 (10 from each year)), (n=30 Females who ran in Fall of 2014 (10 from each year)), (n=30 Males who ran in Fall of 2015 (10 from each year)), (n=30 Females who ran in Fall of 2015 (10 from each year)). I used python's random library to generate pseudorandom numbers in order to select individual cases from the raw data. Below is the nontrivial part of the code (along with description of what every line does) used to generate these four sample frames. It is important to note that this code must be run once for each year, in two ways. It must be run once for each age group, and then that process must be repeated for each of the two years observed.This is because each of the years were provided in separate data files. Once I had all of this data in a nice format, it became trivial to perform statistical tests on. Said statistical tests will be discussed in the analysis section.

Initializes an empty list which will contain the random sample of male times.
```
sampleFrameMale=[]
```
Initializes an empty list for female times
```
sampleFrameFemale=[]
```
Changes context to the list of raw data to be processed
```
for frame in framelist:
```

Creates a temporary list for males who will be included in the sample, but only from one grade/age

```
males=[]
```

Does the same for females

```
females=[]
```

Checks to make sure that only 10 males and 10 females are selected per year

```
while len(males)<10 or len(females)<10:
```

Generates a random integer

```
randomIndex=rdm.choice(xrange(0,len(frame),1))
```

Checks that random value is acceptable

```
if (not pd.isnull(frame["Mile"][randomIndex])) and
frame["Mile"][randomIndex] is not "DNR" and
frame["Mile"][randomIndex] is not "Rx":
```

Checks if gender of randomly selected value is male

```
if frame["Gender"][randomIndex] == "M":
```

Checks if male sample is full

```
if len(males)<10:
```

If not, adds male value to male sample

```
males.append(frame["Mile"][randomIndex])
```

The next 3 lines do the same for females

```
if frame["Gender"][randomIndex] == "F":
    if len(females)<10:
        females.append(frame["Mile"][randomIndex])
```

The last 2 lines return the randomly selected values

```
sampleFrameFemale.append(females)
sampleFrameMale.append(males)
```

The Data:

It was mentioned earlier that the data was quite ugly, for lack of a better term. I noticed a number of inconsistencies which would cause logistical issues. A few of them are as follows. Data entered by different teachers followed different protocols. For example, if a student ran a mile in 7 minutes and 53 seconds, some teachers would record this as “7:53”, while other teachers would record this as “7.53” or “7 53”. This causes an annoying-if solvable- issue, as python will treat something containing “ ” or “:” as a string, while it will treat something containing a “.” as an unsigned floating point number. I fixed this by doing a number of explicit casts to integers at various points in the code. Another issue was that it is hard to compare data to each other when they are in the form of “minutes:seconds.” To address this issue, all data were converted to a single, scalar number of seconds. However, this happened after the random sampling, and so therefore is not shown in the four initial sampling frames, which are shown below. Yet another issue was that the data was “dirty”, or in the wrong format, and my code had to change the format significantly. This was done, though it proved somewhat difficult.

As mentioned, the process shown earlier yielded four separate sampling frames. They are shown below in python stacked list syntax, still stratified, with the sub lists being organized youngest to oldest.

Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.

2015 Male Sample Frame:
[['6:44', '9:49', '5:28', '9:33', '7:50', '6:22', '6:35', '9:49', '6:32', '6:22'], ['8:10', '7.32', '6:23', '6.12', '6.04', '8:56', '5.34', '6:31', '8:11', '7:13'], ['6:35', '6:12', '5:41', 6:54, '6:22', '8:29', '8:29', '11:50', '6:41', '6.19']]

2015 Female Sample Frame:
[['9:41', '10:10', '8:33', '10:24', '7:32', '8:45', '10:18', '9:20', '6:28', '7:36'], ['9.15', '9:34', '8:31', '11:00', '7:41', '8:27', '10:00', '8:25', '9:16', '10:26'], ['8:26', '8:26', '8:50', '11.07', '9:12', '9:12', '9:12', '8:06', '8:14', '10:00']]

2014 Male Sample Frame:
[['6.05', '6:28', '10:07', '9:36', '6:50', '7:55', '7:31', '6.03', '8:25', '10:22'], ['6:32', '6:40', '6:14', '7:32', '8:49', '7:54', '8:13', '6:21', '6:40', '7:53'], ['5:20', '8.08', '12:47', '8:42', '6:39', '6:03', '8:42', '6:02', '6:54', '6:22']]

2014 Female Sample Frame:
[['9.57', '6.57', '8.58', '10:29', '9:30', '9.57', '8:45', '11.11', '9:13', '10:55'], ['9:02', '9:02', '7:38', '6:13', '6:03', '9:02', '8:14', '8:50', '8:50', '8:50'], ['8:57', '10:20', '8:20', '10:58', '13:07', '8:28', '7:20', '7:30', '7:10', '11:27']]

These four sample frames were concatenated in different ways that make varying amount of sense. From these four sampling frames, I examined the following nine samples which I hope are respectively representative of the following distributions:

1. Males who ran in 2015
2. Males who ran in 2014
3. Females who ran in 2015
4. Females who ran in 2015
5. All females
6. All males
7. All runners from 2015
8. All runners from 2014
9. All runners

Conditions-The Hard Part:
The point of doing all of this stuff to the data before actually studying it is that there are a set of conditions the data needs to meet in order for the T-test statistics found to be considered valid, and it was somewhat difficult to meet those conditions. The first condition is randomization, meaning that all data is sampled randomly to eliminate statistical biases. This condition is fulfilled by the fact that I used a random number utility to generate the indices of the individual cases I used in my sample. The second condition, independence in two ways, was not met. Independence is required to make sure the samples are not confounding each other. In this case, however, there are a few reasons I could not ensure that all of the samples are independent. The first main reason is that I do not have names for any of the students on my lists. This, combined with the fact that I am sampling over two years, makes it very difficult to ensure that there are no people in the sample for both years, and if there are, they are in the “all runners” sample twice. This could pose an issue, except for the fact that each run is independent of every other run.what one runner did one year has no effect on what they did the next year, though they are likely very similar. Another issue around independence arises

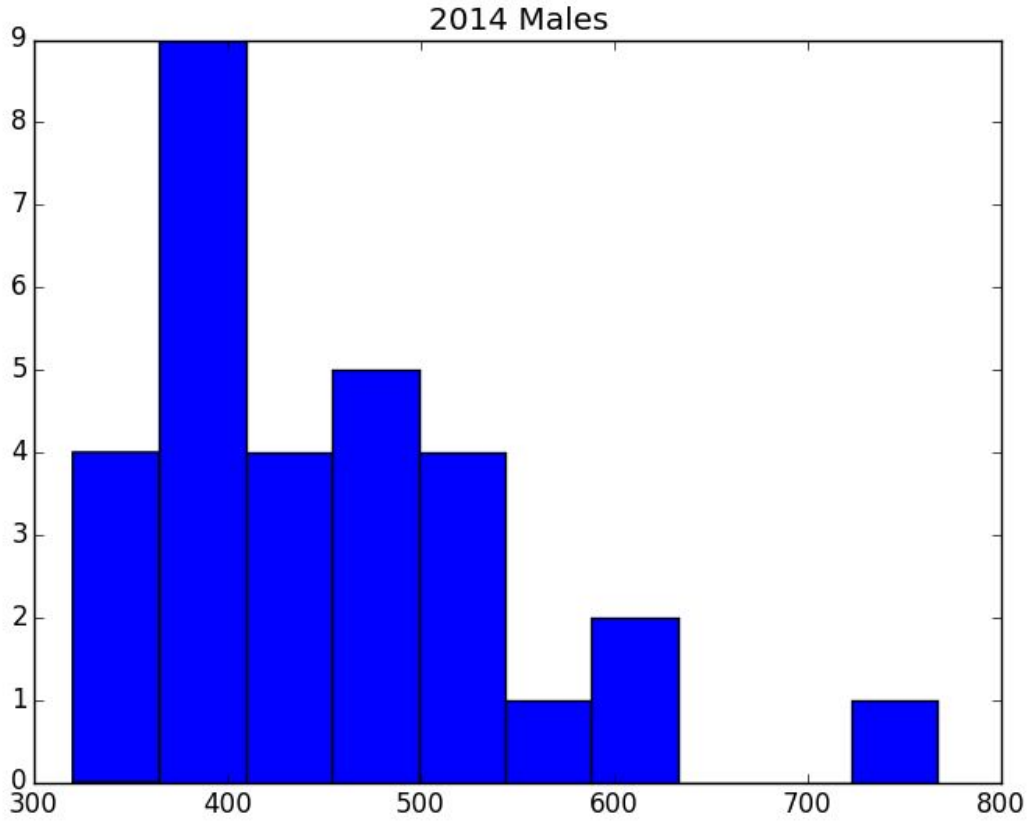
because of the way I am combining my sampling frames. Because they are sometimes combined together in such a way that they share up to 50% of individual cases, it somewhat limits my ability to test. For example, I should not test “2015 runners” against “all runners”, because roughly 50% of the values are shared. However, I can test “All females” against “All males”, due to the fact that people who fit into “all males” definitionally do not fit into “all females.” The third condition is the so-called 10% condition. The 10% condition exists in order to ensure that the sample size is not too large. To examine the point of this, we consider the extreme case in which the sample size is the same as the population size: then there is only one possible sample mean, so the sampling distribution isn't really normal in any meaningful sense. The CLT is an asymptotic result that holds when the population that you are sampling from is infinite (so that your sample size can grow unboundedly). For finite populations, we can sort of wave our hands and pretend that we're actually sampling *with replacement,* which effectively gives us an infinite population to sample with (since we can sample each individual in the population unboundedly many times)--so the CLT holds. If the sample size is small relative to the population size then sampling with and without replacement is almost the same, so we can pretend like we're sampling with replacement even if we're not. But as the sample size grows, sampling without replacement becomes very different from sampling with replacement. However, with my samples of size 30, 60, and 120 at the absolute largest, when the size of the entire population is over 400, it is still okay to have samples of that size.The fifth and final condition is the “nearly-normal condition”, which requires all data summarized by a T-statistic to be unimodal and symmetric, which describes a T distribution. This can be met here because samples are all of size 30 or larger, allowing me to invoke the CLT, which states that samples of sufficiently large size

Analysis:
This large amount of data yielded a lot of information. Believe it or not, I found more data than I have added in this paper. I have attempted to only discuss important analysis, though I have shown each distribution in a separate histogram.

T-test statistics are the primarily used statistics for analyzing quantitative data, and attempting to find the mean of such data. There are two primary operations that are used in this case, and they are a T-confidence interval and a T-test. A T-confidence interval is used to estimate the mean of a population, with a certain amount of confidence, based on a (representative) sample. The equation for a one sample T-confidence interval is as follows: $\overline{X} \pm t^*_{df} * (S/\sqrt{n})$, where X is the sample mean, S is the sample standard deviation, t* is the critical T-test statistic, df is the degrees of freedom (n-1), and n is the sample size. What this means, is that with however much confidence, we can state that the true mean is within $t^*_{df} * (S/\sqrt{n})$ of $\overline{X}$. This is useful for finding (or at least approximating) the true values of means based on smaller samples. I had a computer do all of the actual data processing, and that is included in my code. Below is each of the nine distributions along with sample statistics and 95% T-intervals.

Male Runners From 2014:

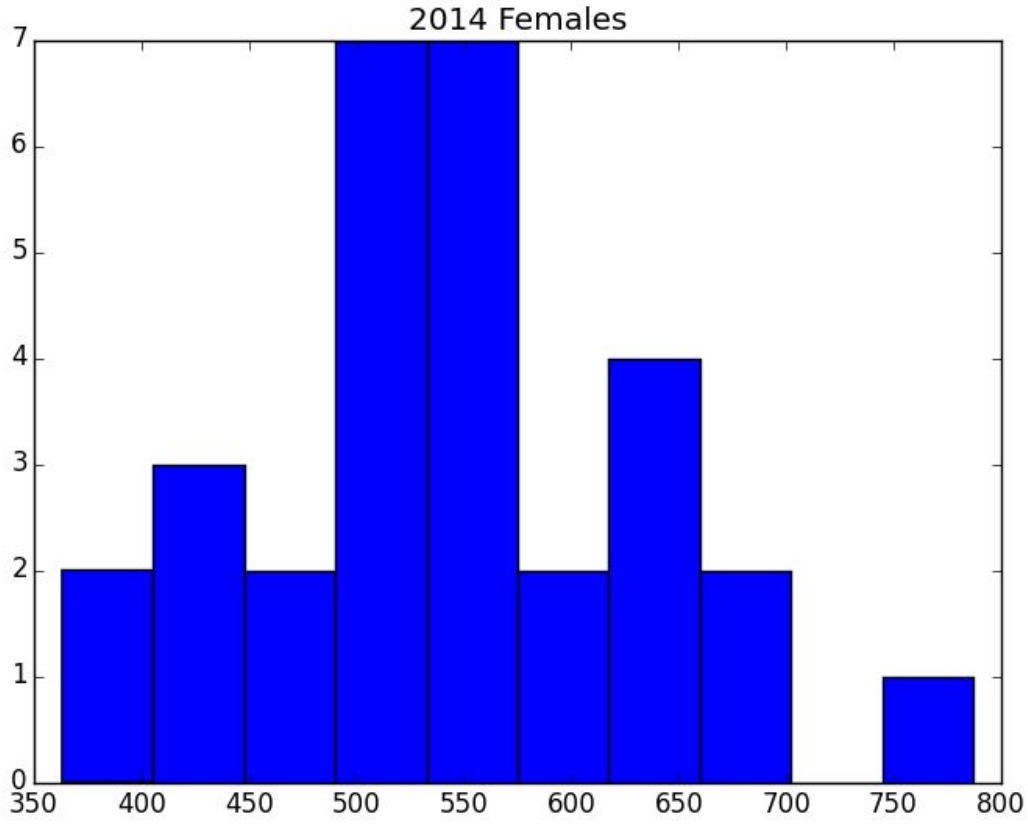
Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.



Mean: 455.6333s. StdDev:95.600s. 95% Confidence interval:[419.935s-491.330s].

Female Runners from 2014

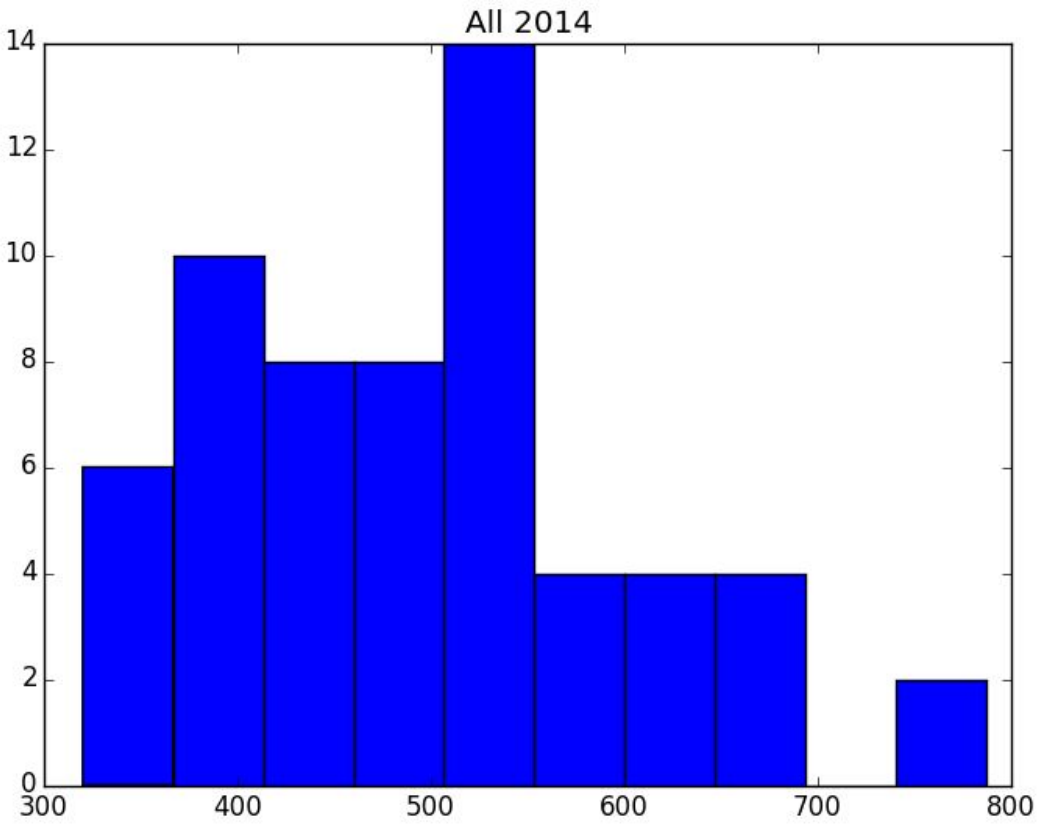
Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.



Mean: 542.433s. StdDev:94.0211s. 95% Confidence interval:[507.325s-577.541s].

Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.

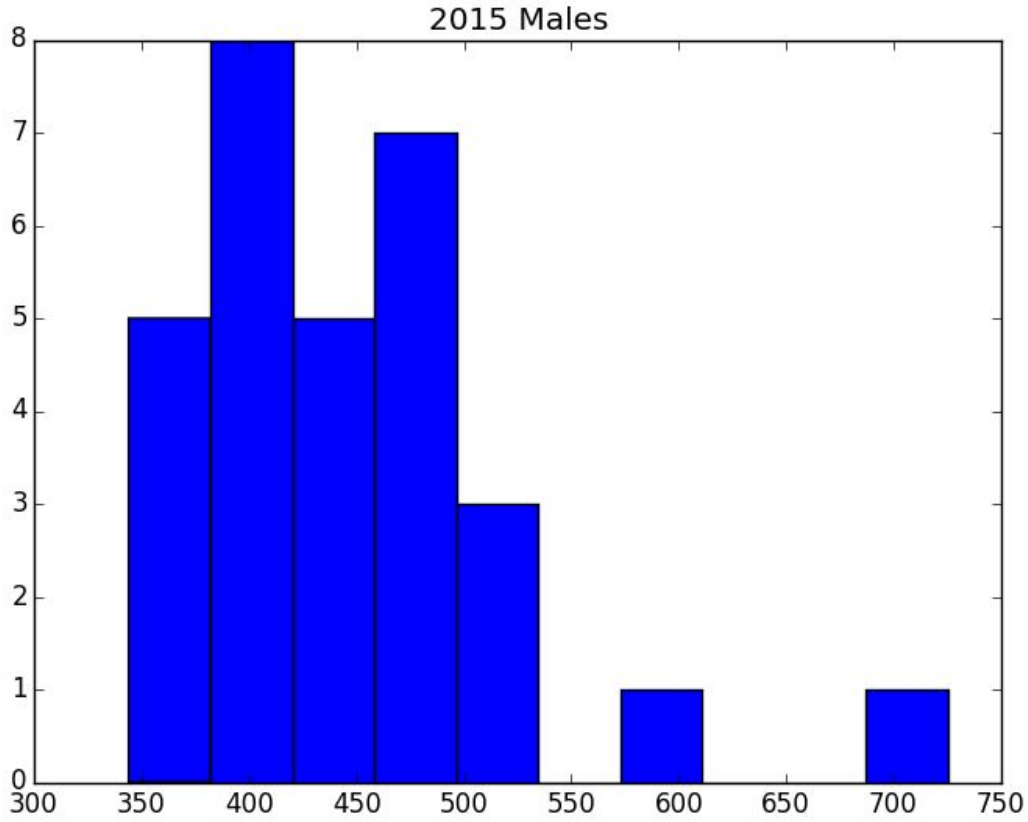
All Runners from 2014:



Mean: 499.033s. StdDev:104.275s. 95% Confidence interval:[472.096s-525.97s].

Male Runners from 2015:

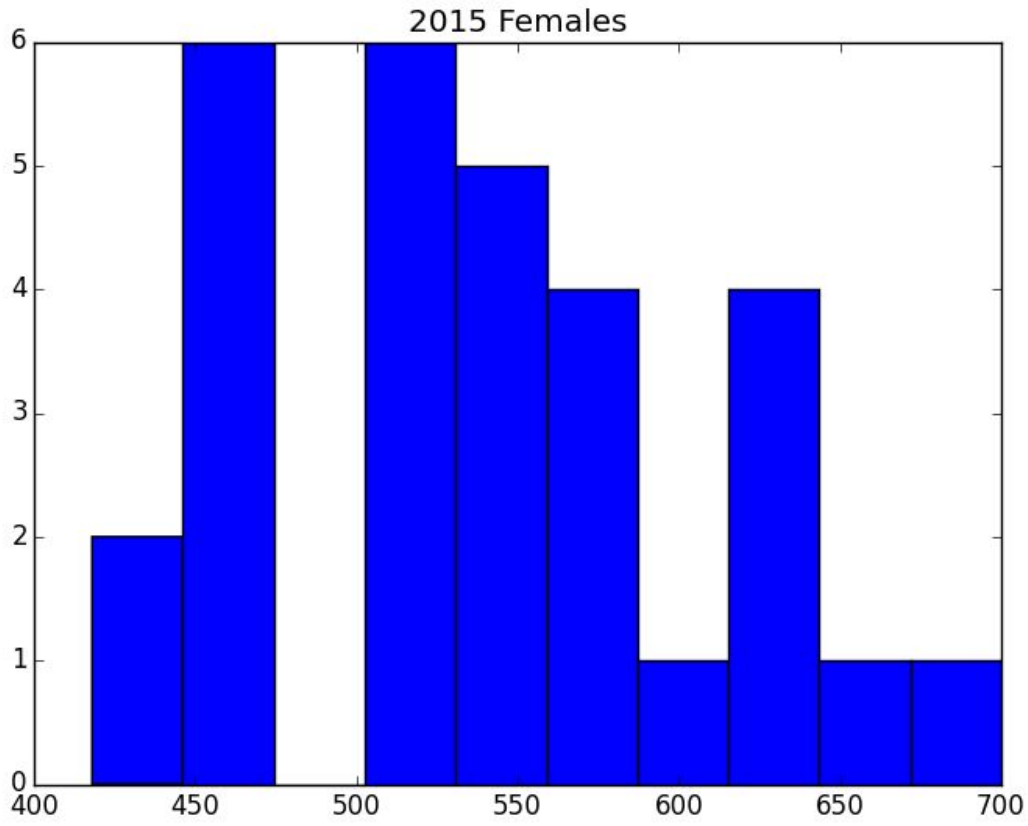
Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.



Mean: 446.333s. StdDev:74.267s. 95% Confidence interval:[418.601s-474.065s].

Female Runners from 2015:

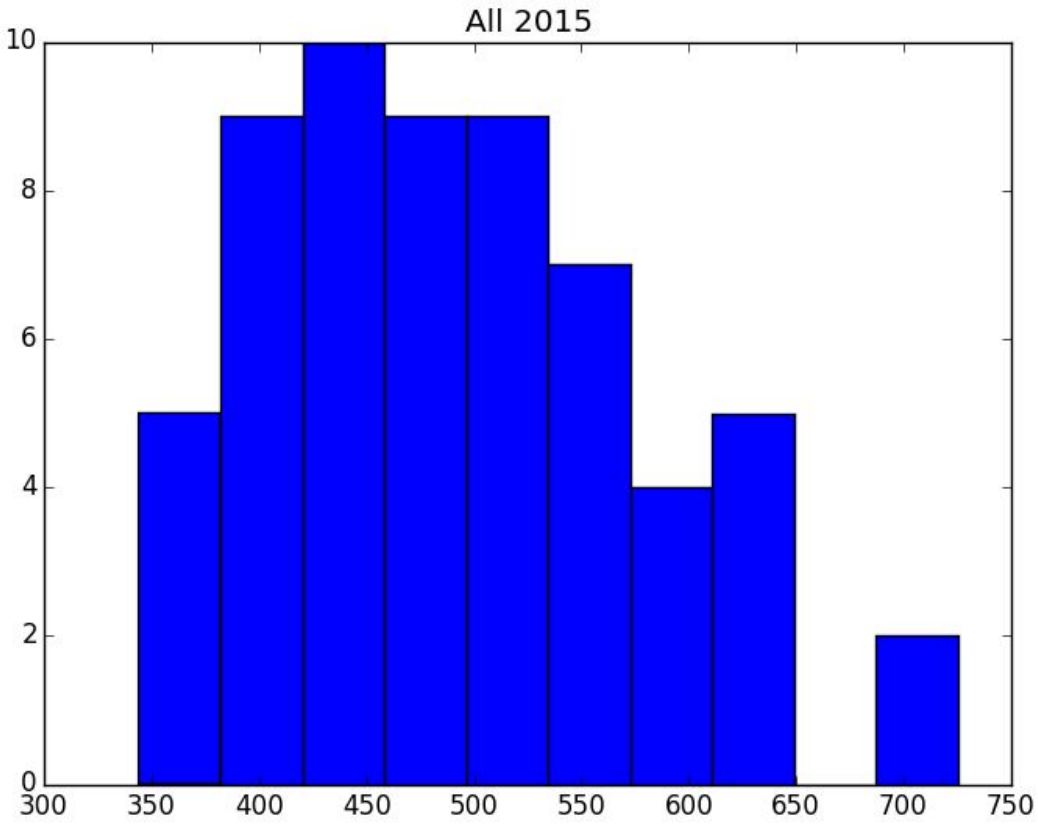
Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.



Mean:539.666. StdDev:70.037s. 95% Confidence interval:[513.514s-565.81s].

All 2015 Runners:

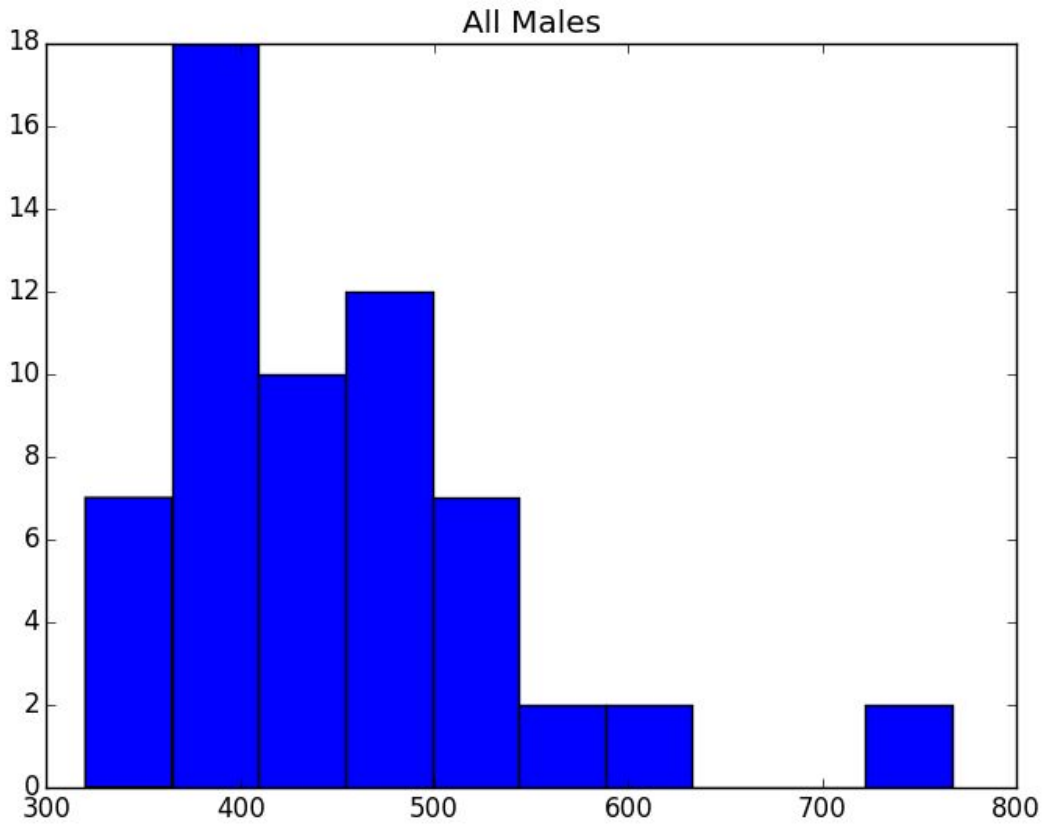
Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.



Mean:493.0s. StdDev:85.954s. 95% Confidence interval:[470.795s-515.204s].

Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.

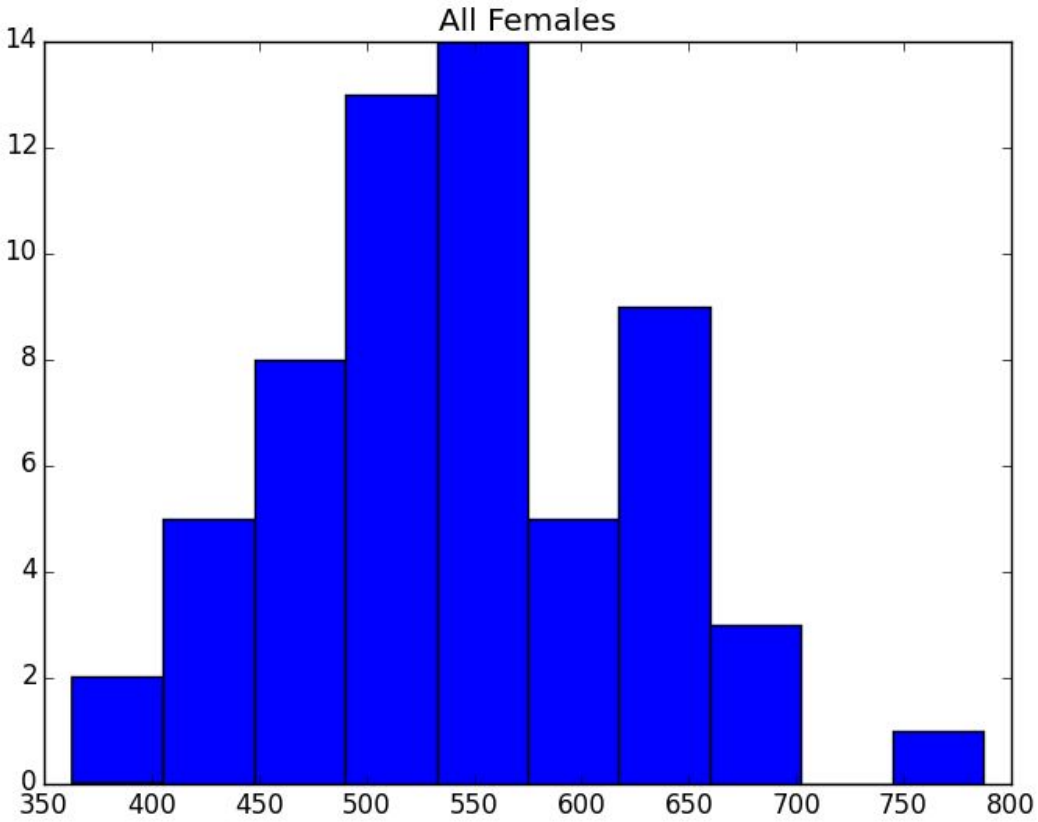
All Male Runners:



Mean:450.983. StdDev:85.727s. 95% Confidence interval:[428.837s-473.129s].

Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.

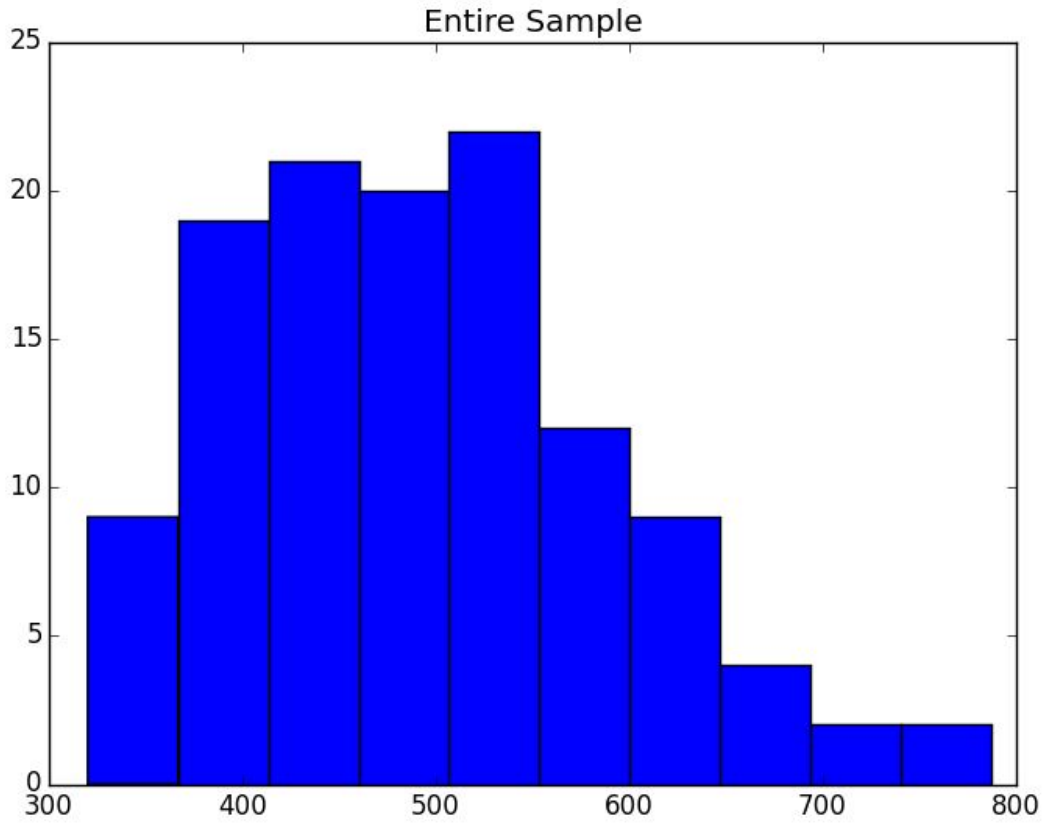
All Female Runners:



Mean:541.05. StdDev:82.912s. 95% Confidence interval:[519.631s-562.468s].

Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.

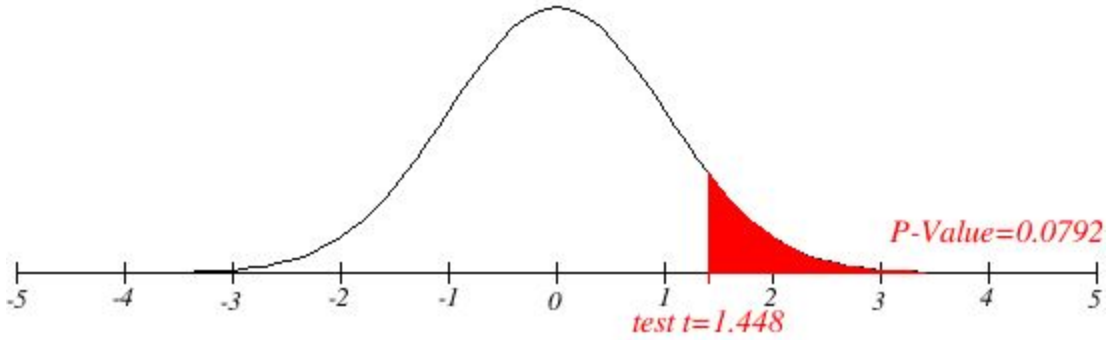
All Runners:



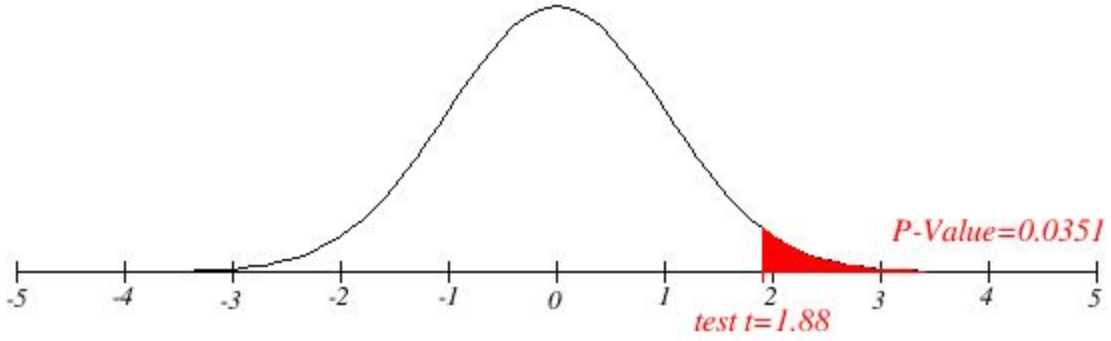
Mean:496.017. StdDev:95.602s. 95% Confidence interval:[478.735s-513.297s].

Where T-confidence intervals are used to estimate distribution means based on samples, T-statistic tests are used to compare distributions via samples. They come in two flavors: one sample, and two sample. A one sample T-test is used to compare the mean of a population to a known constant. For me, this is useful in the case of comparing data to the nationally accepted averages for high school students. The formula for a 1-sample T-test is: $t_{df} = \frac{\overline{X} - \mu}{\frac{s}{\sqrt{n}}}$, where t is the T-test statistic, which can be used to find a conditional probability, $\overline{X}$ is your projected mean, μ is the null hypothesis mean, S is the sample StdDev, df is the degrees of freedom (n-1), and n is the sample size. The conditional probability found by using the T test statistic can be interpreted as follows: the probability that, given the null is true, your sample contains the results you found or more extreme. Using this statement, combined with an “alpha cutoff” or significance value, you can make statements about the comparison between the mean of the population and the supplied constant without ever seeing the true mean of the population itself. The national average mile time for a high-school age student, according to the national physical fitness program, is 430 seconds for males, and 631 seconds for females. In my case, I ran every single population against the national average for both males and females, and using an alpha

cutoff of 0.05, here are some of the results that were interesting. With a T-test statistic of +1.443, and a p-value of .0792 there is not statistically significant evidence that males in 2014 ran slower than the national average for males. Df is n-1, in this case, 29

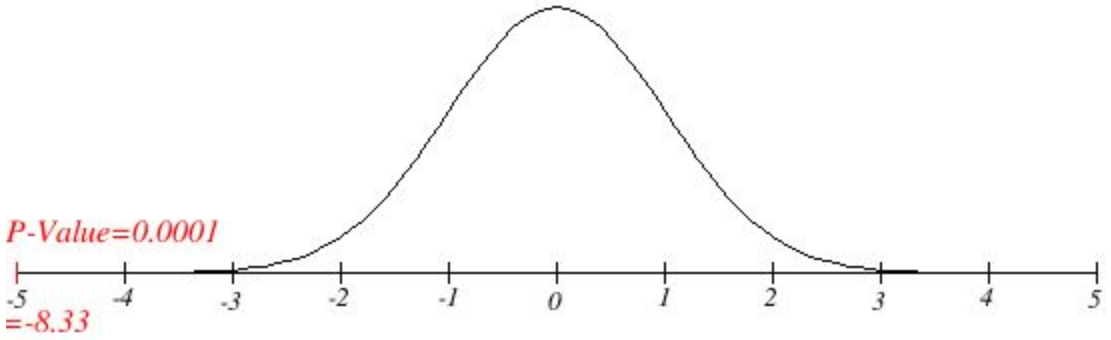


However, with a T-test statistic of +1.88, and a p-value of .0351, there is statistically significant evidence that the entire subset of males at the lab school typically run slower(longer times) than the national average for males of 430 seconds. Df is n-1, or 59.

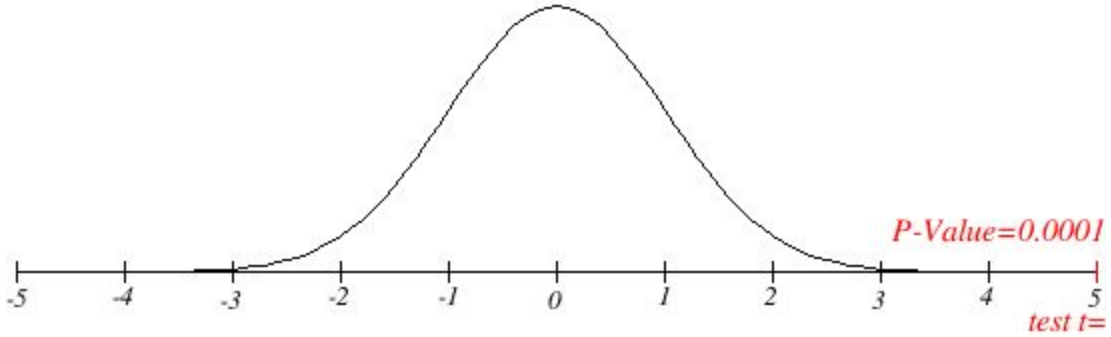


Ben Glick
May 3, 2016
Final Project: AP Stats
J. Maguire, Inst.

Females at Lab run much faster than the national average, in all 3 female only categories, most prominently in the all females sample, where there was a t test statistic of -8.33, with a p value of 1.49x10^-11, or essentially 0. Df is equal to n-1, in this case 59.



The mean of all runners at lab over these two years is very clearly slower than the male national average, with a T-test statistic of +7.53, and a p-value of 1x10^-11. Degrees of freedom is 119.

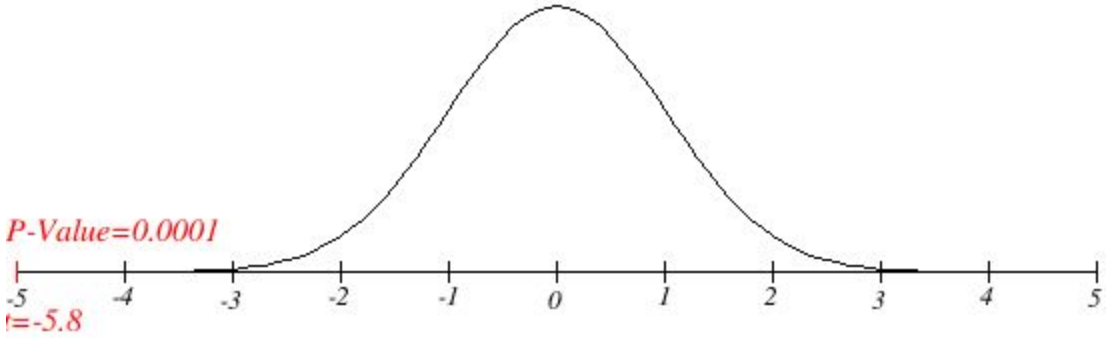


There were 14 1-sample T-tests run, and their output is attached with with the raw data.

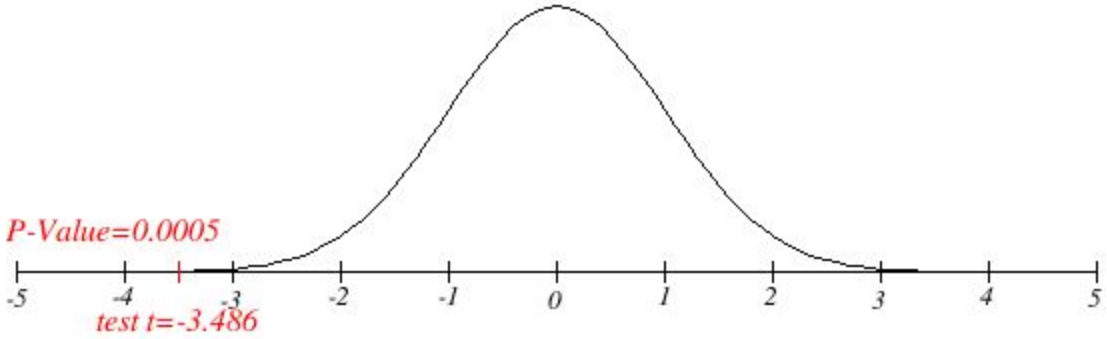
In addition to 1-sample T-tests, there are also 2-sample T-tests, which work in a very similar way to 1-sample ones, except that instead of comparing one population to a constant, one uses a 2-Sample test to compare one population to another population, all without ever seeing the entire population, and simply using smaller samples. The formula for the 2-Sample T-test statistic is: $t_{df} = \frac{\overline{X}_1 - \overline{X}_2}{\sqrt{\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}}}$ where t is the T-test statistic, which can be used to find a conditional probability, $\overline{X}$ is your projected mean,X1 and X2 are the two sample means, S1 and s2 are the two sample StdDevs, df is the degrees of freedom, and n1 and n2 are the sample sizes. In order to find degrees of freedom with this test, you must use this equation: $DF = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\left[\frac{1}{n_1 - 1}\left(\frac{s_1^2}{n_1}\right)^2 + \frac{1}{n_2 - 1}\left(\frac{s_2^2}{n_2}\right)^2\right]}$

The conditional probability found by using the T test statistic can be interpreted as follows: the probability that, given the null is true, your sample contains the results you found or more extreme. The T and P values are essentially the same as before, and the alpha cutoff can be described similarly to the way we did it with 1-sample T-tests, it's just a little more complicated to get there. I performed quite a few 2-sample T-tests, because in my code, it compares every sample to every other sample, essentially, without asking. All of those T-tests' computer output is included with the raw data, but only a couple are discussed here. As mentioned earlier, some of these tests are statistically unsound, due to the independence condition. I have omitted those from the paper, but they are still in the raw output.

The distribution of all males at Lab has faster times than the distribution of all females. This t-test is statistically valid because male and female samples are definitionally independent. The T-test statistic here is -5.8007 (male times are a lower number of seconds, hence the -), and the p-value is 5.6x10^-8. Because that p-value is lower than the alpha-cutoff, of 0.05, there is statistically significant evidence that males typically have run faster than females during the last two years at Lab. Degrees of freedom is given by 117.9 or so.



Another interesting T-test showed that in 2014, the typical male ran much faster than the typical female, with a T-test statistic of -3.4860, and a p-value of .00045. An example of a test that does not make any sense to run would be all 2015 runners vs all runners. This does not make sense because the two populations share roughly 50% of values, making the populations essentially homogeneous, and the T-test statistic essentially invalid. Degrees of freedom here is approximately 57.9.



All of the other T-tests are available in the raw output file.

Conclusion:
In conclusion, T tests are a very valuable tool to use for comparisons of quantitative data, given that one uses proper precautions, such as always making sure conditions are met before using statistical results. They can tell you a lot about a population without the unnecessary hassle of performing a census. Upon application of T-tests to Lab mile times, I was able to learn a lot about the distribution of running times at Lab without actually having to go through every single runner at the school.