Data-based Statistical Decision Model

Lecture 1 - Optimal Prediction

Sungkyu Jung

Lecture overview

1. Optimal Prediction
2. The case of simple linear regression
3. Model checking and inference
4. Data visualization and wrangling
5. Multivariate linear models
6. Model checking and inference
7. Model evaluation and variable selection
8. Smoothing methods
9. Simulations and bootstraps
10. Logistic regression, GLMs and GAMs
11. More general models

Lab overview

1. R basics, RStudio, and Rmarkdown
2. Data visualization
3. Data Wrangling

Today

Why try modeling at all?

Regression analysis is about investigating quantitative, predictive relationships between variables. It's about situations where there is some sort of link, tie or relation between two (or more) variables, so if we know the value of one of them, it tells us something about the other. The concrete sign of this is that knowledge of one variable lets us predict the other - predict the target variable better than if we didn't know the other. Pretty much everything we are going to do in this class is about crafting predictive mathematical models, seeing whether such models really have any predictive power, and comparing their predictions. Before we get into the issues of statistics and data analysis, it will help us to think what optimal prediction would look like, if we somehow knew all the probability distributions of all our variables.

Statistical Prediction and the Optimal Linear Predictor

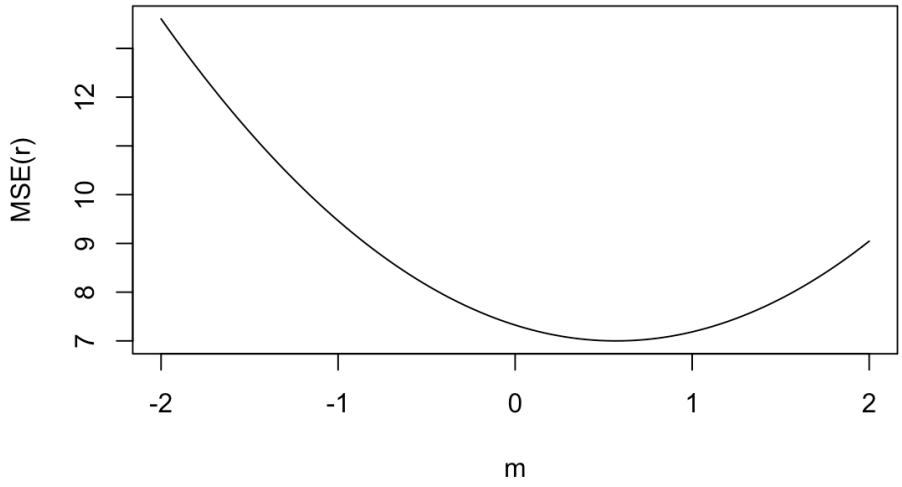
Predicting a Random Variable from Its Distribution

Suppose we want to guess the value of a random variable Y. Since we don't feel comfortable with the word "guess", we call it a "prediction" instead. What's the best guess we can make?

What is the optimal point prediction for Y ? Depends on how we measure error. If we consider mean squared error,

$$MSE(r) = E[(Y - r)^2],$$

then the optimal prediction is $r = E(Y)$



Predicting One Random Variable from Another

Now imagine we have two random variables, say X and Y. We know X and would like to use that knowledge to improve our guess about Y. Our guess is therefore a function of x, say $m(x)$. We would like $E[(Y - m(X))^2]$ to be small.

The optimal function just gives the optimal value at each point:

$$\mu(x) = E[Y|X = x]$$

This $\mu(x)$ is called the (true, optimal, or population) regression function (of Y on X).

The Optimal Linear Predictor

- Unfortunately, in general $\mu(x)$ is a really complicated function
- “what is the best prediction we can make which is also a simple function of x?”
- “What is the optimal prediction we can make which is linear in X?” That is, we restrict our prediction function $m(x)$ to have the form $b_0 + b_1 x$.
- Thus minimize

$$MSE(b_0, b_1) = E(Y - (b_0 + b_1 X))^2 ��$$

The optimal linear predictor, or the optimal regression line, is

$$\mu(x) = \beta_0 + \beta_1 x,$$

where

$$\beta_0 = E[Y] - \beta_1 E[X],$$

$$\beta_1 = Cov[X, Y]/Var[X]$$

Important Morals

1. We did NOT assume that the relationship between X and Y really is linear.
2. The best linear approximation to the truth can be awful. (Imagine $E[Y|X = x] = e^x$, or even $= \sin x$.)
3. No assumptions on distributions.
4. No assumption on the fluctuations of Y (No need to be Gaussian, or symmetric)
5. In general, changing the distribution of X will change the optimal regression line.
6. At no time did we have to assume that X came before Y in time, or that X causes Y

Reminder: Basic Probability theory

(see blackboard)

Reminder: Estimation and quantifying uncertainty

(see blackboard)

Understanding random variables and their distributions through simulations

1. Distribution of a random variable

What does it mean that a random variable X follows, say, Exponential(1) distribution?

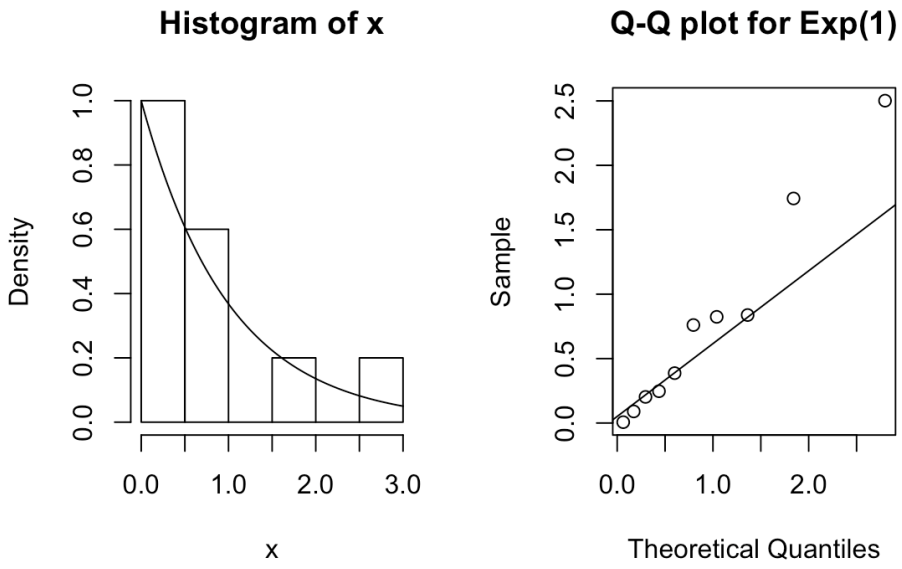
- One answer: If you randomly sample observations from the population

  $$X_i \sim \text{ i.i.d. } \text{Exp}(1),$$

  then the observations $\{X_1, \ldots, X_n\}$ are distributed according to the Exponential distribution.

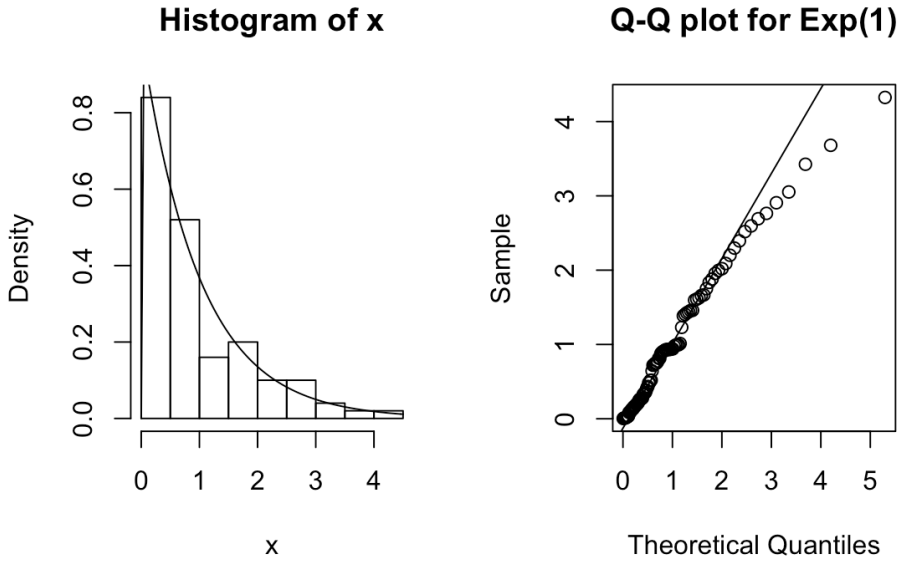
- See this for increasing sample size n.

For $n = 10$:

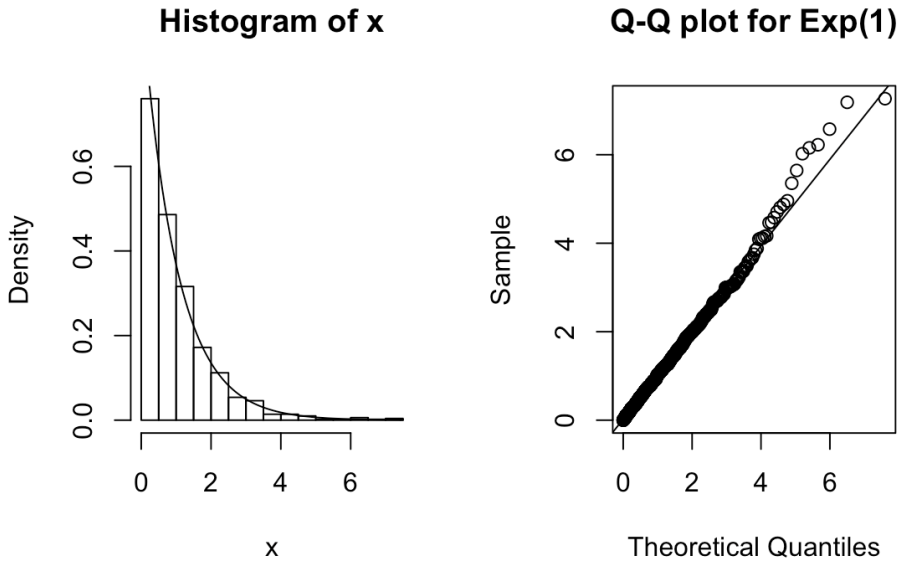


Compare the histogram with overlaid theoretical Exponential(1) density $f(x) = \exp(-x), x > 0$.

For $n = 100$:



For $n = 1000$:



- The density function of X can be well-approximated by plotting a histogram of i.i.d. sample $X_1, \ldots, X_n$.

- The larger sample size, the better the approximation.

2. Distribution of a statistic

Let $X_1, \ldots, X_n$ be a random sample of size n from $\mathrm{Exp}(1)$ population. There are a number of statistics of interest.

1. $\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$
2. $S_n = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X}_n)^2$
3. $\sqrt{\bar{X}_n}$
4. $V_n = \bar{X}_n / S_n$
5. $\tilde{X}_n$, the median of the sample
6. The "average" of mean and median: $(\bar{X}_n + \tilde{X}_n)/2$

Sampling distributions of statistics

What is the distribution of the statistic $\bar{X}_n$?

- Can be evaluated theoretically:
- Gamma$(n, 1/n)$

How about the distribution of the statistic $\sqrt{\bar{X}_n}$?

- Can also be evaluated theoretically. It follows an unconventional distribution called generalized gamma distribution.

How about the distribution of the statistic $\tilde{X}_n$ (the median)?

- Theoretical evaluation is quite challenging
- Instead, approximate!

Approximating the sampling distribution of $\tilde{X}_n$

Recall:

The density function of $\tilde{X}_n$ can be well-approximated by plotting a histogram of i.i.d. sample $(\tilde{X}_n)_1, \ldots, (\tilde{X}_n)_N$.

The larger "sample size N", the better the approximation.

Note that there are now two "sample sizes":

- n: the sample size of the random sample $X_1, \ldots, X_n$ (that constitutes one sample)
- N: the number of samples (randomly drawn for the purpose of simulation)

Let the sample size be $n = 30$. Draw one sample (of size n):

```
n <- 30
x <- rexp(n = n, rate = 1) # observe a random sample of size n
x
```

```
##  [1] 0.30447606 1.14753465 1.00485984 0.08488515 2.88345132 1.05489760
##  [7] 0.84521039 0.29779700 3.37661500 0.17608630 2.40964947 0.69408198
## [13] 0.51781766 0.42957991 0.07436716 0.14244205 2.09198433 1.67990872
## [19] 0.57616717 0.88671457 0.53962325 1.59292151 0.97313280 0.55973481
## [25] 0.54327417 0.35832394 1.25865852 2.42970798 0.45463414 0.05713204
```

and compute the statistic:

```
Median <- median(x)
Median
```

```
## [1] 0.6351246
```

Now draw N many samples (each of size n).

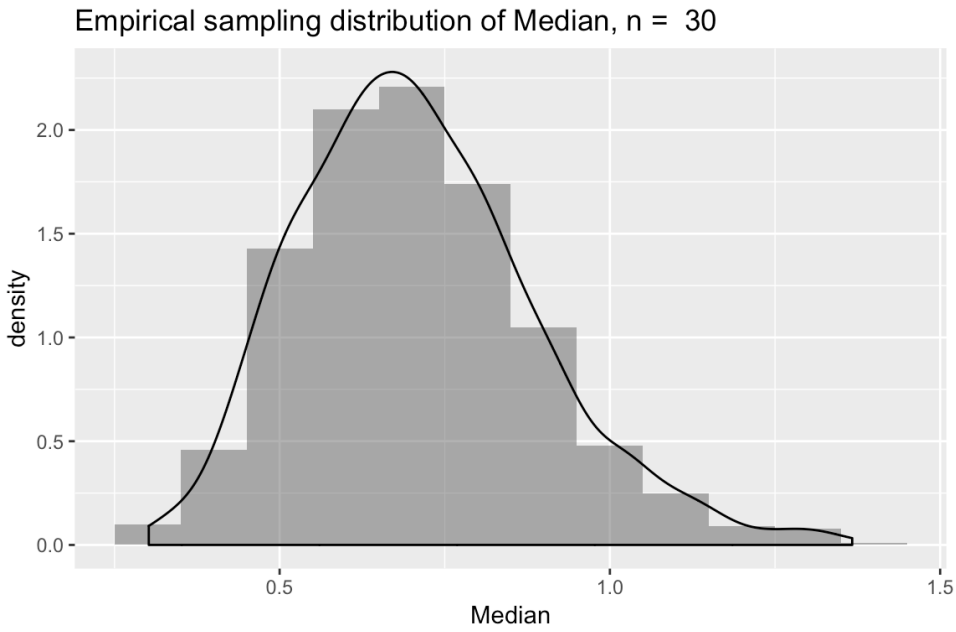
```
n <- 30   # sample size
N <- 1000 # number of random samples
Median <- vector()
for(i in 1:N){
  x <- rexp(n = n, rate = 1)  # observe a random sample of size n
  Median[i] <- median(x)      # compute the statistic
}
str(Median)
```

```
##  num [1:1000] 0.79 0.514 0.879 1.082 0.657 ...
```

```
summary(Median)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.3019  0.5788  0.6912  0.7079  0.8160  1.3669
```

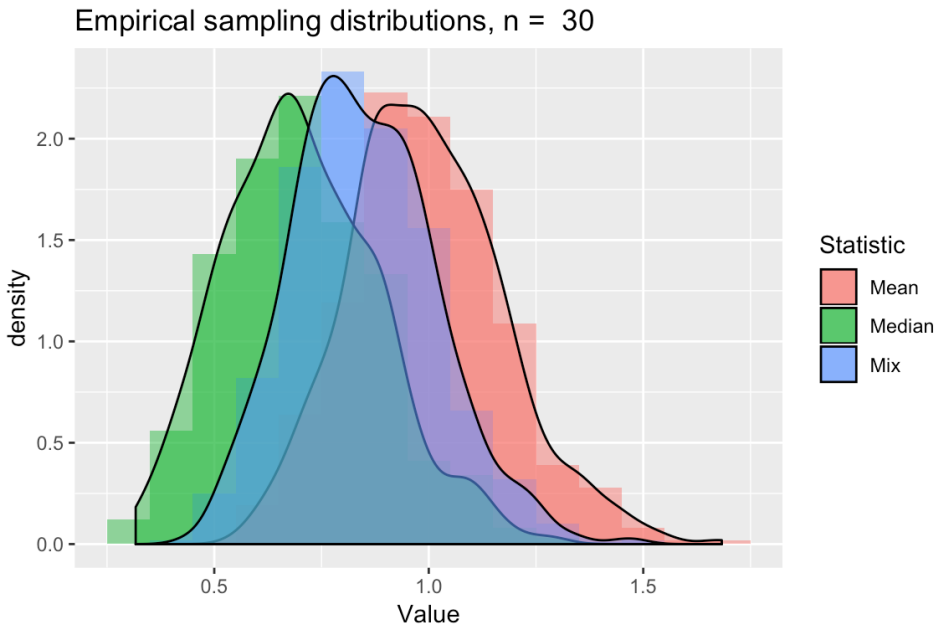
Plot a histogram of N-many observed $\tilde{X}_n$'s, with "smoothed histogram"" overlaid.



We are now ready to compare the sampling distribution of the mean $\bar{X}_n$, median $\tilde{X}_n$ and their average $M_n = (\bar{X}_n + \tilde{X}_n)/2$, obtained from a random sample of size $n = 30$ following $\mathrm{Exp}(1)$.

Recall that it is very difficult to theoretically evaluate the densities of $\tilde{X}_n$ and M_n. But we can approximate them.

```
n = 30   # sample size
N = 1000 # number of random samples
Values = matrix(ncol = 3, nrow = N)
for( j in 1:N){                 # Repeat N times
   xi = rexp(n,rate = 1)        # observe a random sample of size n
   S1 = c(mean(xi), median(xi)) # compute the statistics; mean, median
   Values[j,] = c(S1, mean(S1)) #                         and the average
}
```



3. Approximating probabilities

Throughout, we will use the example of computing $P(X \in A)$, for $X \sim N(0,1)$ and $A = (0,1)$.

Direct calculation (if applicable)

Let $X \sim N(0,1)$. For any interval $A = (a,b)$, $a < b$, computing $P(X \in A)$ amounts to either

- numerically computing $\int_a^b \phi(x)dx$;
- evaulating $\Phi(a)$ and $\Phi(b)$ from the normal table;
- or using a computer software to directly evaluate $\Phi(a) - \Phi(b)$.

See some of these options in R, for $(a,b) = (0,1)$.

Computing the numerical integration

```
a = 0 ; b = 1
p1 = integrate(dnorm, lower =  a, upper = b, mean = 0 ,sd = 1)
```

Evaluating the CDF difference

```
p3 = pnorm(b) - pnorm(a)
```

Result:

	Probability
Numerical integration	0.3413447
By CDF difference	0.3413447

큰 모집단에서 무작위로 뽑은 표본의 평균이 전체 모집단의 평균과 가까울 가능성이 높다(출처: 위키백과)

Approximation by law of large numbers

The probability can also be approximated, by an application of the law of large numbers.

- First we obtain replicates of X by randomly drawing

$$X_1, \ldots, X_n \sim \text{ i.i.d. } N(0,1).$$

- Define Y_i, for each i, satisfying

$$Y_i = \begin{cases} 1, & \text{if } \mathrm{X_i} \in \mathrm{A}; \\ 0, & \text{if } \mathrm{X_i} \notin \mathrm{A}. \end{cases}$$

- Then $Y_i \sim$ i.i.d. Bernoulli(p) with $p = E(Y_1) = P(X_1 \in A)$

- The law of large numbers says that as $n \to \infty$.

$$\frac{1}{n}\sum_{i=1}^{n} Y_i \to P(X_1 \in A),$$

in probability.

Thus, as n increases, we expect that the (observed) $\bar{Y}_n$ converges to $P(X_1 \in A)$.

Let us approximate the value of $P(X \in (0,1))$, $X \sim N(0,1)$.

```
xi = rnorm(20000, mean = 0, sd = 1)
yi = xi < b & xi > a
nvec = c(50,100,1000,5000,10000,20000)
p4 = vector()
for(ii in 1:6){
   n = nvec[ii]
   p4[ii] = mean(yi[1:n])
}
```

	Probability
50	0.3800000
100	0.3800000
1000	0.3280000
5000	0.3422000
10000	0.3430000
20000	0.3406000

	Probability
Numerical integration	0.3413447

This principle of approximation can be applied to evaluate any probability or expected value. For example,

1. $X \sim \text{Poisson}(4)$. Approximate $P(X < 5)$.
2. X_i i.i.d. $\sim \text{Poisson}(4)$. Let $Y = X_1/(X_2 + X_3 + 1)$. Approximate $P(Y < 1/2)$.
3. Approximate $E(Y)$.

> Try these in your free time!

4. Understanding asymptotics through simulations

The three fundamental asymptotic results below are basic building blocks of mathematical statistics. For i.i.d. $X_1, \ldots, X_n$, with $E(X_1^2) < \infty$, we have

- The law of large numbers
  - $\bar{X}_n \to E(X_1)$ in probability as $n \to \infty$
- The central limit theorem 동일한 확률분포를 가진 독립 확률 변수 n개의 평균의 분포는, n이 충분히 크다면 정규분포에 가까워진다는 정리이다(출처: 위키백과)
  - $\sqrt{n}(\bar{X}_n - E(X_1)) \to N(0, Var(X_1))$ in distribution as $n \to \infty$
- The delta method for, e.g., $g(x) = \sqrt{(}x)$
  - $\sqrt{n}(\sqrt{\bar{X}_n} - \sqrt{E(X_1)}) \to N(0, (g'(E(X_1)))^2 Var(X_1))$ in distribution as $n \to \infty$

In the following, we will check that these theories are realized for the Exponential(1) population. For this, we need to consider the sampling distribution of $\bar{X}_n$ for several values of n. Each sample $X_1, \ldots, X_n$ will be repeatedly observed (N times) to obtain the approximate sampling distribution of $\bar{X}_n$.

Let's take an increasing sequence of sample sizes $n = 5, 10, 25, 100$, and simulate.

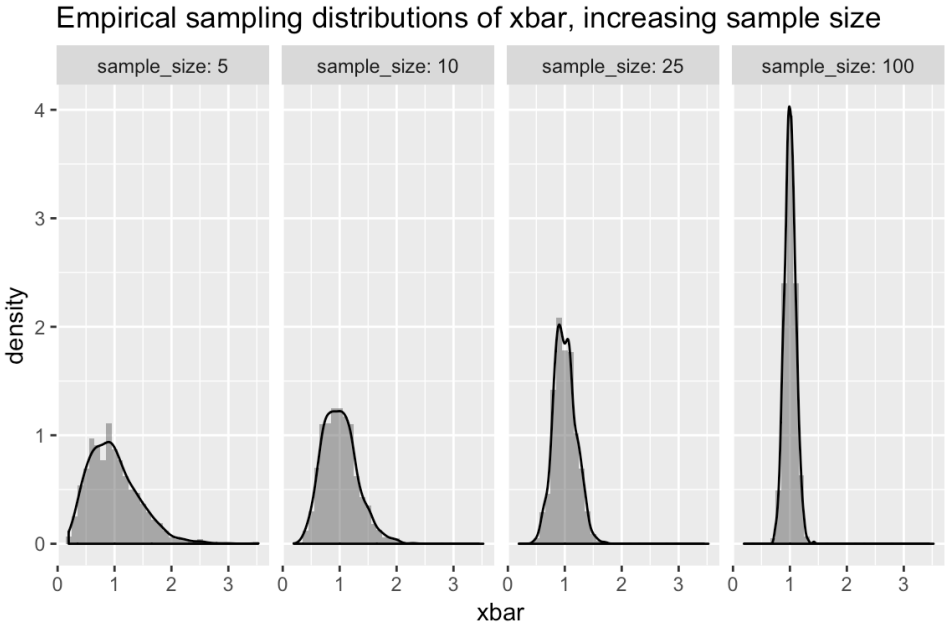
```
nseq = c(5, 10, 25, 100) # sample size sequence
N = 1000                 # number of random samples

df = data.frame()
for(icase in 1:4){       # for each choice of sample size
   n = nseq[icase]
   xbarvec = vector()
   sqxbarvec = vector()
   for( j in 1:N){       # repeat N times
   x = rexp(n,rate = 1)  # observe a random sample of size n
   xbarvec[j] = mean(x)            # compute the mean
   sqxbarvec[j] = sqrt(mean(x))    #         and its sq. root
   }
   df <- rbind(df, cbind(xbarvec, sqxbarvec, rep(n,N)) )
}
```

```
## 'data.frame':    4000 obs. of  4 variables:
##  $ xbar             : num  1.285 1.335 1.333 0.939 0.908 ...
##  $ sqrt_xbar        : num  1.134 1.156 1.154 0.969 0.953 ...
##  $ sample_size      : num  5 5 5 5 5 5 5 5 5 5 ...
##  $ sample_size_factor: Factor w/ 4 levels "n = 10","n = 100",..: 4 4 4 4 4 4 4 4 4 4 ...
```

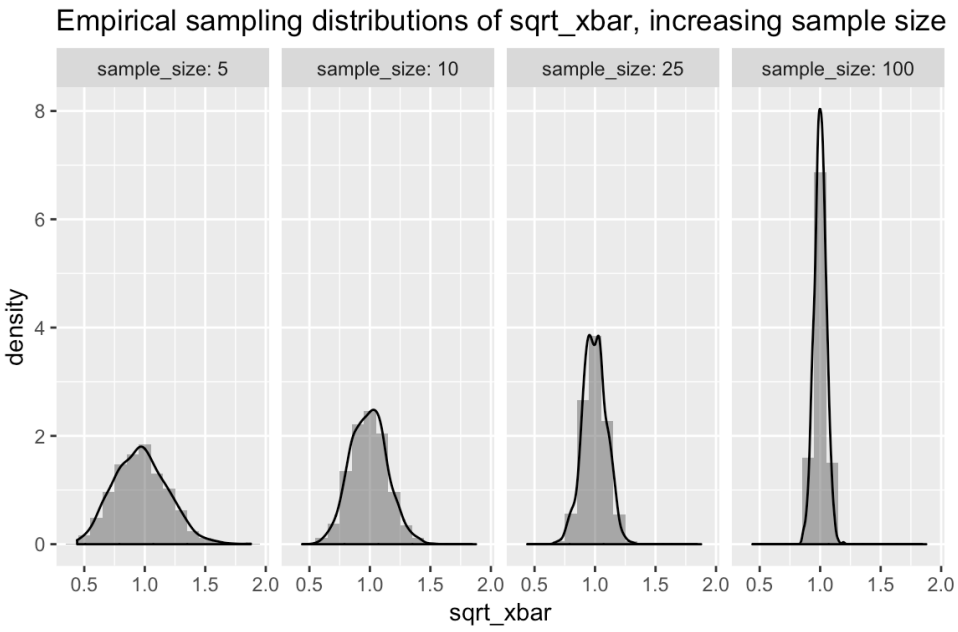
Law of large numbers

Do you see that $\bar{X}_n \to E(X_1) = 1$ in probability?



sample size가 작을 땐 1 주변으로 넓게 퍼져 있다가 sample size가 커질수록 1주변으로 모여드는 것을 확인할 수 있다. 대부분의 값들이 1과 가까워졌으므로 그래프가 높게 솟아있다.

Do you see that $\sqrt{\bar{X}_n} \to \sqrt{E(X_1)} = 1$ in probability?

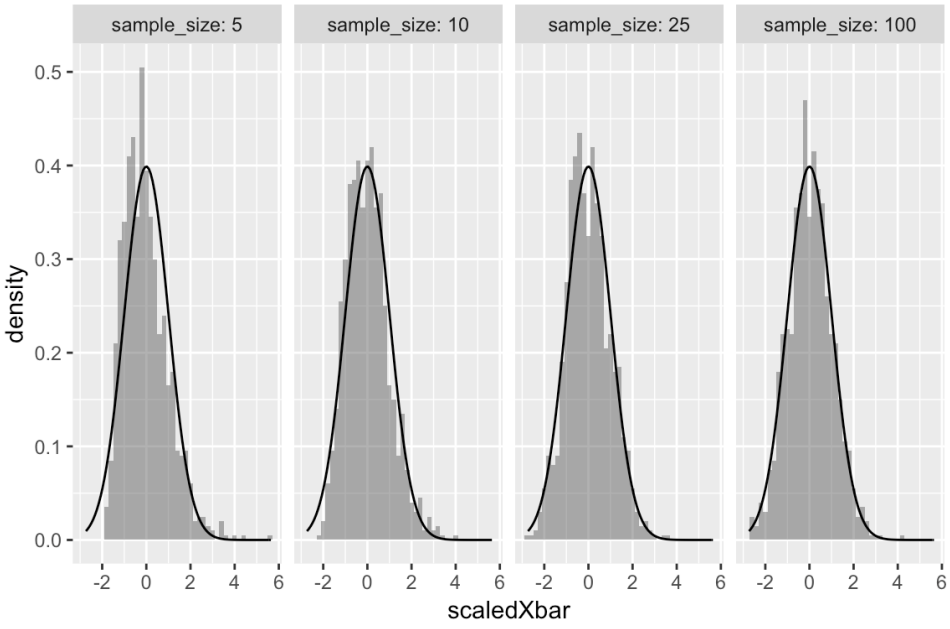


Central limit theorem

First, we translate and scale $\bar{X}_n$:

```
scaledXbar = sqrt(df$sample_size) * ( df$xbar - 1 );
df = cbind(df, scaledXbar)
```

Now, overlay the histrograms with theoretical standard normal densities. Do you see $\sqrt{n}(\bar{X}_n - E(X_1)) \to N(0, Var(X_1))$ in distribution?



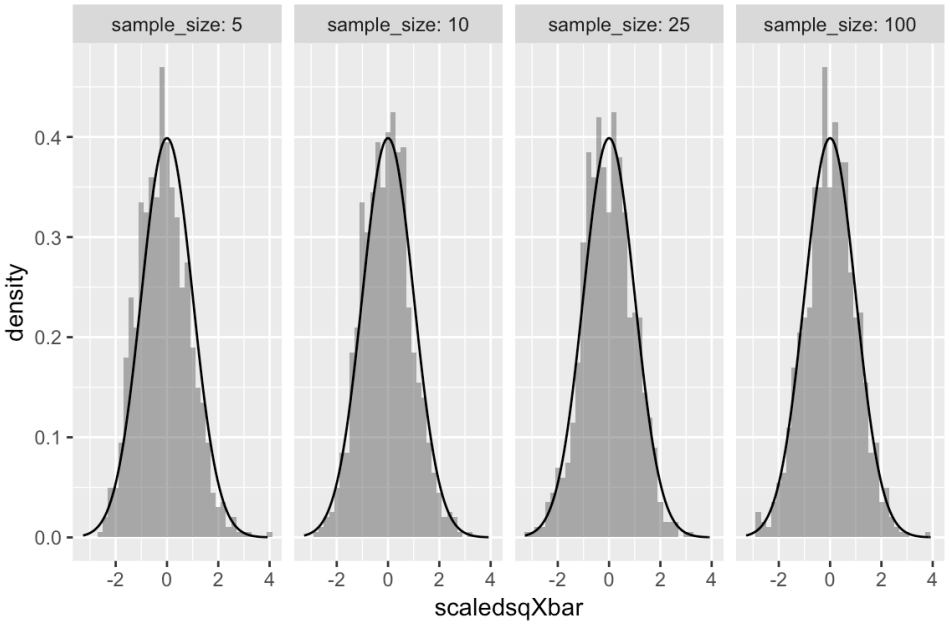
점점 정규분포 형태에 가까워지는 것을 확인할 수 있다.

Delta method

First, we translate and scale $\sqrt{\bar{X}_n}$:

```
scaledsqXbar = sqrt(df$sample_size) * ( df$sqrt_xbar - 1 ) * 2;
df = cbind(df, scaledsqXbar)
```

Now, overlay the histrograms with theoretical standard normal densities. Do you see $\sqrt{n}(\sqrt{\bar{X}_n} - \sqrt{E(X_1)}) \to N(0, [g'E(X_1)]^2 Var(X_1))$ in distribution?



Remark

- Distribution of ANY random variable (either a single random variable or a statistic) can be *numerically* approximated by simulation
- Probabilty of any event can be *numerically* approximated
- Expected values of random variables can be *numerically* approximated
- Simulations are thus used to probe otherwise-difficult-to-evaluate quantities

Intermission

Introducing statistical modeling

Let's start this off with a motivating example. We'll begin by loading some data which comes from the Bureau of Economic Analysis, on the economic output of cities in the U.S. (https://www.bea.gov/index.htm (https://www.bea.gov/index.htm)).

```
bea = read.csv("bea-2006.csv")
```

```
dim(bea)
```

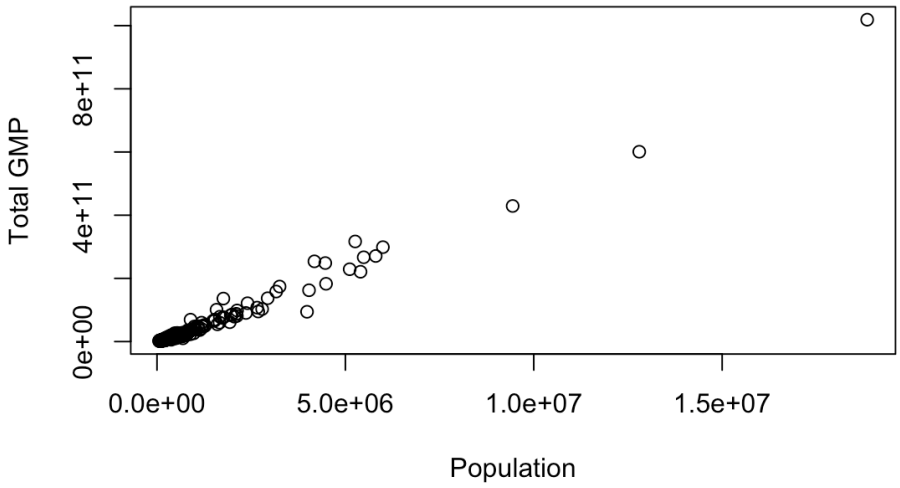
```
## [1] 366   7
```

```
head(bea,2)
```

```
##           MSA pcgmp    pop finance prof.tech     ict management
## 1 Abilene, TX 24490 158700  0.0975        NA 0.01621         NA
## 2   Akron, OH 32890 699300  0.1294    0.0544      NA    0.05431
```

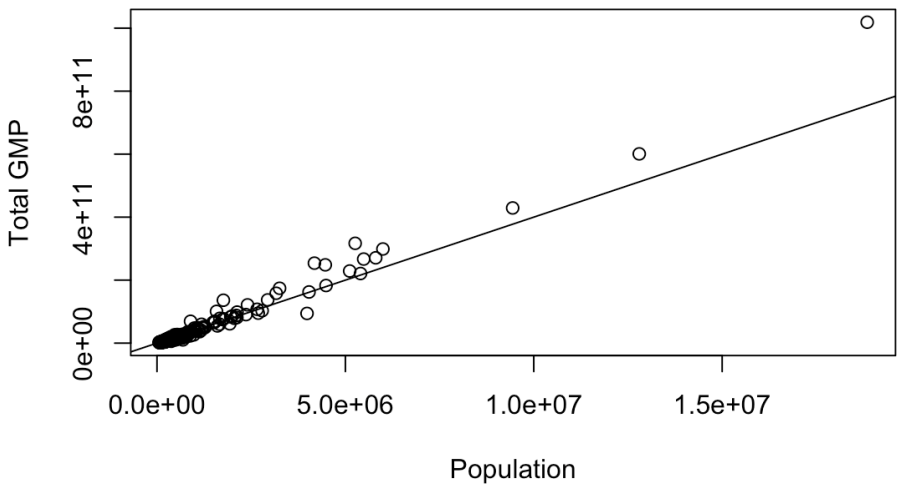
Let's add a new column, which records the total GMP, by multiplying the output per person by the number of people:

```
bea$gmp <- bea$pcgmp * bea$pop
plot(gmp ~ pop, data = bea, xlab = "Population", ylab = "Total GMP")
```



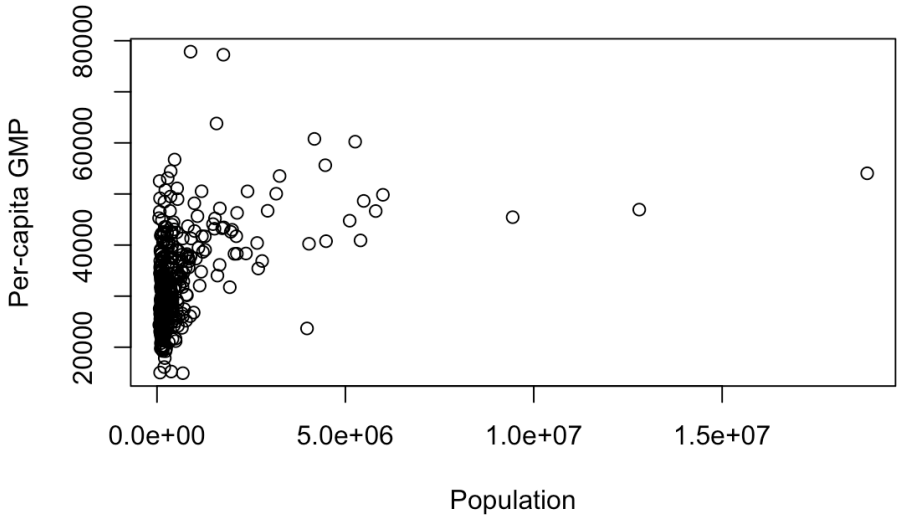
Larger cities have larger total economic outputs.

```
plot(gmp ~ pop, data = bea, xlab = "Population", ylab = "Total GMP")
abline(a = 0, b = 40000)
```



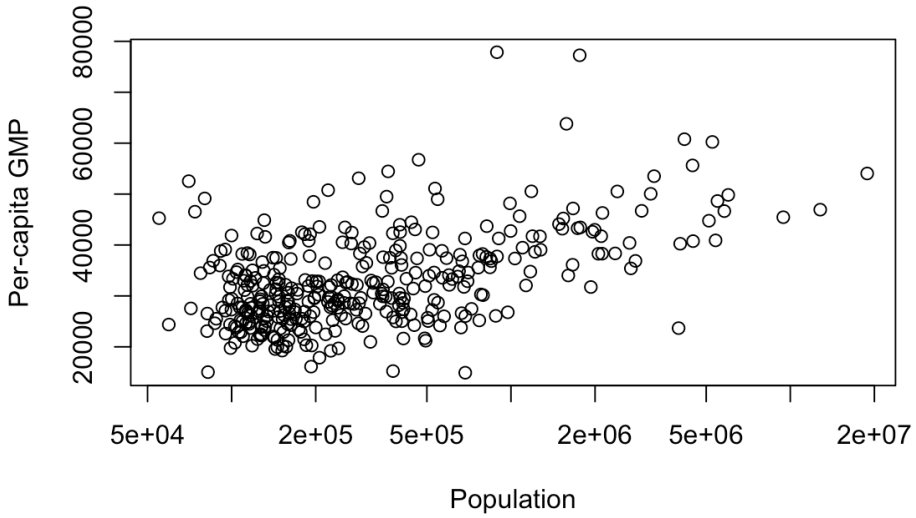
This isn't bad at all, but it looks like it's systematically too low for the larger cities. This is suggestive that there may be differences between the economies of large and small cities. Let's explore this by looking at the per-capita figures.

```
plot(pcgmp ~ pop, data = bea, xlab = "Population", ylab = "Per-capita GMP")
```



At this point, it becomes annoying that the larger cities in the US are so much larger than the small ones. By using a linear scale for the horizontal axis, we devote most of the plot to empty space around New York, Los Angeles and Chicago, which makes it harder to see if there is any trend. A useful trick is to switch to a logarithmic scale for that axis, where equal distances correspond to equal multiples of population.

```
plot(pcgmp ~ pop, data = bea, xlab = "Population", ylab = "Per-capita GMP",
    log = "x")
```

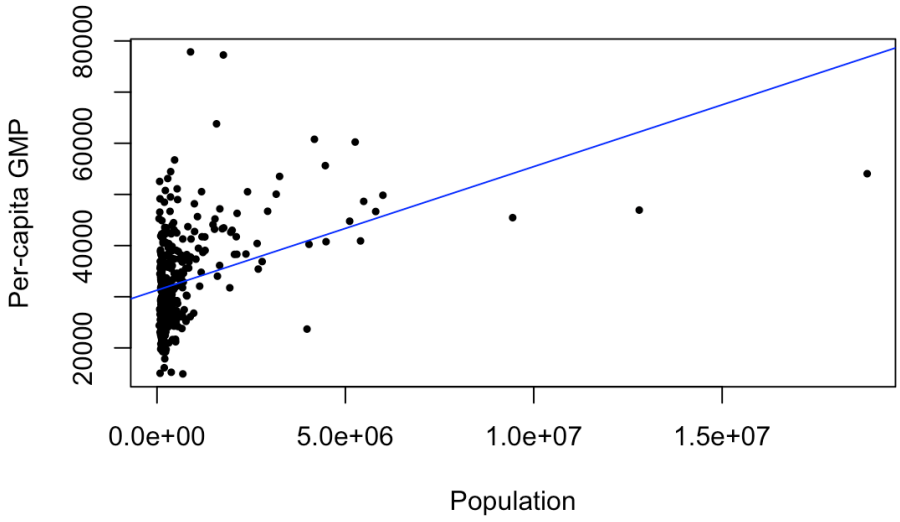


Let's now calculate our first regression line. R has a function for estimating linear models, with which we'll become very familiar:

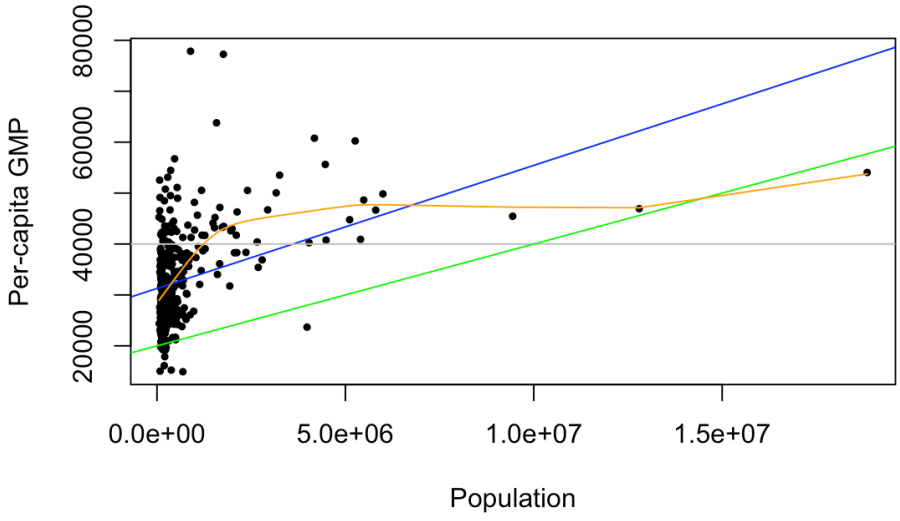
```
lm(pcgmp ~ pop, data = bea)
```

```
##
## Call:
## lm(formula = pcgmp ~ pop, data = bea)
##
## Coefficients:
## (Intercept)          pop
##   3.128e+04    2.416e-03
```

```
plot(pcgmp ~ pop, data = bea, xlab = "Population", ylab = "Per-capita GMP",
    pch = 19, cex = 0.5)
abline(lm(pcgmp ~ pop, data = bea), col = "blue")
```



But why should we believe that line? R does it, but I hope by this point in your life you don't think "the computer says it, so it must be right" is ever a good idea. Why prefer that blue line over this grey line, or the green one, or for that matter over the orange curve?



Why do we want to fit any lines here at all?

What are statistical models for?

Summaries

In many situations, we'd rather ignore all of that precise detail and get away with a summary; we might want to compress the 732 numbers into just an intercept and a slope that describe the general trend or over-all shape of the data.

Anything you can calculate from the data could, in principle, be used as a summary. Even if the calculation was originally inspired by doing some sort of statistical inference, every statistic can be a descriptive statistic.

Inference

If we want to go beyond describing, summarizing or compressing the data, we enter the realm of **inference** — we try to reach out and extend our knowledge from the data we have, to other variables we have not measured, or not measured so directly. This is inherently somewhat *risky, imprecise, and uncertain*. In statistics, we aim *not only to draw such inferences, but to say something about the level of risk, imprecision, and uncertainty which accompanies them*.

Inferences can be subject to uncertainty

- When the data is just a sample of a larger population, and we want to extrapolate from the sample to the population.
- If this is not the case, why does it seem wrong to say that the slope of the optimal linear predictor is exactly 3.1277574×10^4, 0.0024162?

Why it's reasonable to do statistical inference on a complete data set

- Account for measurement error
  - systematic
  - statistical
- Account for accidental values; aiming at the general, underlying trends
- Each observation has fluctuations, the distribution of the accidents is stable + When we try to quantify uncertainty, we want to know how different our calculations could have been.

Statistical Models

- To say anything useful here, we will need to make assumptions to say anything about how different the data could, plausibly, have been.
- These take the form of statistical models or probability models, through random variables.
- Specifying a statistical model
  - what the random variables are
  - what the restrictions for them are
  - how they relate to each other

Example: Conceivable statistical models for the BEA data

1. $X \sim N(6.81 \times 10^5, 2.42 \times 10^{12})$; $Y|X \sim N(4.00 \times 10^4, 8.50 \times 10^7)$; X independent across cities; Y independent across cities given their X's.
2. $X \sim N(\mu_X, \sigma_X^2)$ for some mean μ_X and variance σ_X^2 ; $Y|X \sim N(4.00 \times 10^4, 8.50 \times 10^7)$; Y independent across cities given their X's.
3. $X \sim N(\mu_X, \sigma_X^2)$ for some mean μ_X and variance σ_X^2; $Y|X \sim N(4.00 \times 10^4, \sigma_Y^2)$ for some variance σ_Y^2 ; Y independent across cities given their X 's.
4. distribution of X unspecified; $Y|X \sim N(4.00 \times 10^4, \sigma Y2)$ for some σY2 ; Y independent across cities given their X's.
5. distribution of X unspecified; $Y|X \sim N(\beta_0 + \beta_1 x, \sigma_Y^2)$ for some β_0,β_1,σ_Y^2; Y independent across cities given their X's.
6. distribution of X unspecified; $E[Y|X = x] = \beta_0 + \beta_1 x$ for some β_0 and β_1; $Var[Y|X = x] = \sigma_Y^2$ for some σ_Y^2 ; Y independent across cities given their X 's.
7. distribution of X unspecified; $E[Y|X = x] = \beta_0 + \beta_1 x$ for some β_0 and β_1 ; Y uncorrelated across cities given their X's.

Models, inference and model checking

- The stronger the assumptions we make, the stronger the inferences we can draw, *if the assumptions are true*
- When confronting a data analysis problem, you first need to formulate a statistical model.
- Once you have a model, the two key tasks are inference within the model, and checking the model.
- Inference: estimation, or prediction, or hypothesis testing, or confidence intervals
  - formalized, mathematical, rigorous (as good as the modeling assumptions that they're based on)
- Model checking: seeing whether the assumptions are really true.
  - much less formalized, mathematical and algorithmic than inference
  - There is no *The Right Way To Do It*

Simple Linear Regression

The simple linear regression model is a model with two random variables, X and Y, where we are trying to predict Y from X. Here are the model's assumptions:

1. The distribution of X is unspecified, possibly even deterministic;
2. $Y|X = \beta_0 + \beta_1 x + \varepsilon$,where ε is a noise variable;
3. ε has mean 0, a constant variance σ^2, and is uncorrelated with X and uncorrelated across observations.

The assumptions I have just laid out, while they are non-trivial because they could be violated (and are, in many situations), are still strong enough to let us get a start on inference.

Estimation

Remember we saw last time that the optimal linear predictor of Y from X has slope

$$\beta_1 = Cov[X, Y]/Var[X].$$

But both $Cov[X, Y]$ and $Var[X]$ are functions of the true distribution. Rather than having that full distribution, we merely have data points, say $(x_1, y_1), (x_2, y_2), \ldots (x_n, y_n)$. How might we estimate β_1 from this data?

An obvious approach would be to use the data to find the sample covariance and sample variance, and take their ratio. the sample variance of X is

$$s_X^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

while the sample covariance is

$$c_{XY} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

So the sample slope (empirical slope) is

$$\hat{\beta}_1 = \frac{c_{XY}}{s_X^2}$$

Inference: What can we say more about the model?

- Is $\hat{\beta}_1 = \beta_1$?
- When can we expect $\hat{\beta}_1$ be close to β_1?
- What is the expected magnitude of "$\hat{\beta}_1 - \beta_1$"?
- Will stronger assumptions (e.g. Gaussian) give a stronger statement?