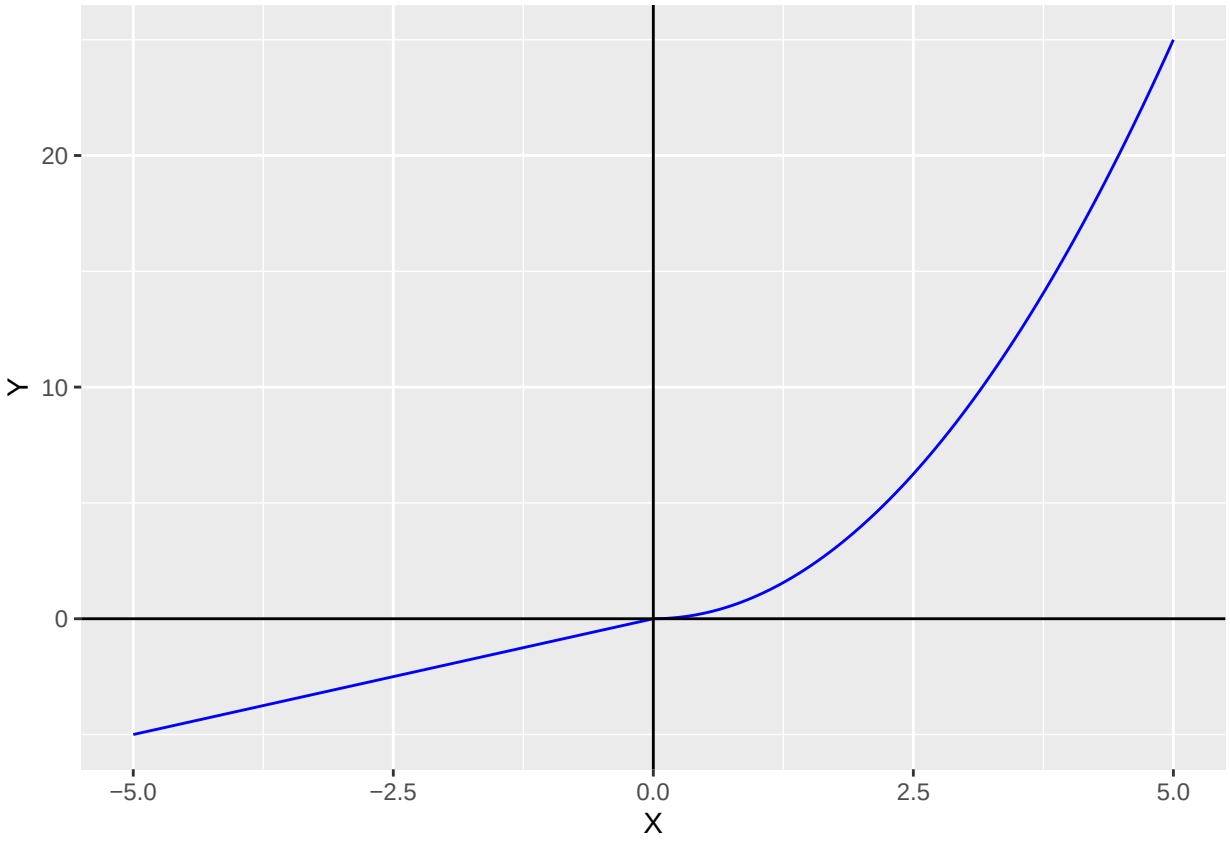
HW 2

2023-02-05

P1

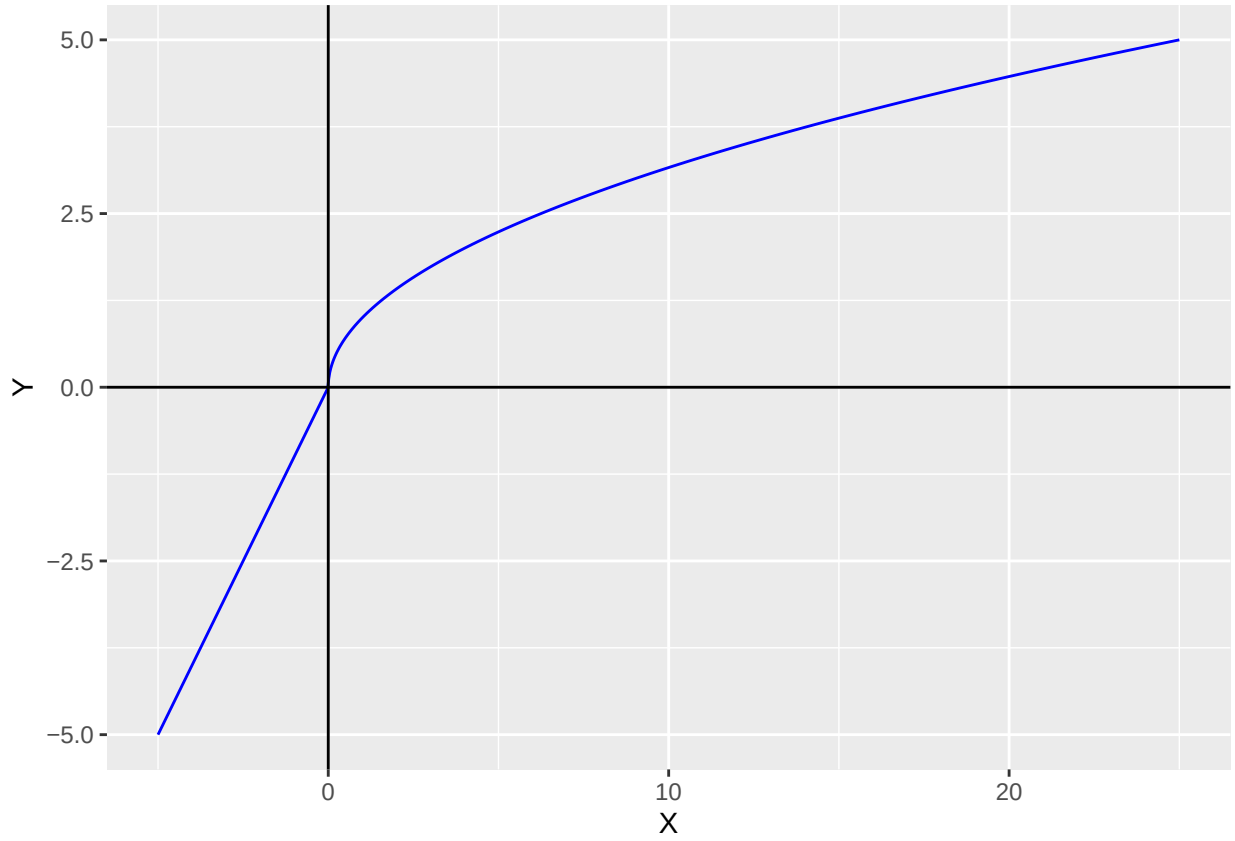
1

```
ya <- tibble(X = seq(-5,5, length = 1000)) %>% #prob 1.1
  mutate(Y = case_when(X < 0 ~ X,
                       X >= 0 ~ X^2)) %>%
  ggplot() +
  geom_line(aes(x=X,y=Y), color = "blue") +
  geom_hline(yintercept = 0) +
  geom_vline(xintercept = 0)


ya
```



2

```
ya_inv <- tibble(X = seq(-5,25, length = 1000)) %>% # prob 1.2
  mutate(Y = case_when(X < 0 ~ X,
                       X >= 0 ~ X^0.5)) %>%
  ggplot() +
  geom_line(aes(x=X,y=Y), color = "blue") +
  geom_hline(yintercept = 0) +
  geom_vline(xintercept = 0)


ya_inv
```



P2

1

- It can describe random events where we cannot (yet) possibly elucidate all possible deterministic factors that go into an event occurring or not. Because of this we can model the likelihood of some "random" event occurring or not using probability.

2

- Modeling Weather Outcomes.

3

- Modeling the change in position with respect to time of a ball tossed straight into the air.

P3

1

$P(A \cap B) = P(A) + P(B) - P(A \cup B)$

2

- When A and B are disjoint events relative to each other. As such, both conditional probabilities would equal zero.

3

- $P(A) \leq P(B)$

P4

1

1. The number of thunderstorm asthma events in the Minneapolis-St. Paul metro area from 2007-2018.
2. The number of asthma related emergency room visits in the Minneapolis-St. Paul metro area from 2007-2018.
3. The amount of pollen in the air in the Minneapolis-St. Paul metro area from 2007-2018.
4. The number of lightning counts in the Minneapolis-St. Paul metro area from 2007-2018.

P5

1

- $P(X > a) = P(Y > a)$

P6

1

```
univ_prob1 <- tibble(w = seq(0,1, length = 1000), #6.1
                     x = w^2) %>%
  mutate(inside = (x >= 0.2 & x <= 0.3) | (x >= 0.9 & x <= 1)) %>%
  count(inside) %>%
  mutate(prob = n/sum(n))

answer_p6.1 <- pull(univ_prob1[2,3])
answer_p6.1
```

```
## [1] 0.153
```

2

```
univ_prob2 <- tibble(w = seq(0,1, length = 1000), #6.2
                     y = (1-w)^2) %>%
  mutate(inside = (y >= 0.2 & y <= 0.3) | (y >= 0.9 & y <= 1)) %>%
  count(inside) %>%
  mutate(prob = n/sum(n))
answer_p6.2 <- pull(univ_prob2[2,3])
answer_p6.2
```

```
## [1] 0.153
```

3

```
#6.3
funk <- function(w) {
  w^2 + 1
}

probability <- function(a,b,X) {
  prob <- tibble(w = seq(0,1, length =1000),
                 Fx = X(w)) %>%
    mutate(inside = Fx >= a & Fx <= b) %>%
    count(inside) %>%
    mutate(prob = n/sum(n))
    return(pull(prob[2,3]))
    #return(prob)
}
```