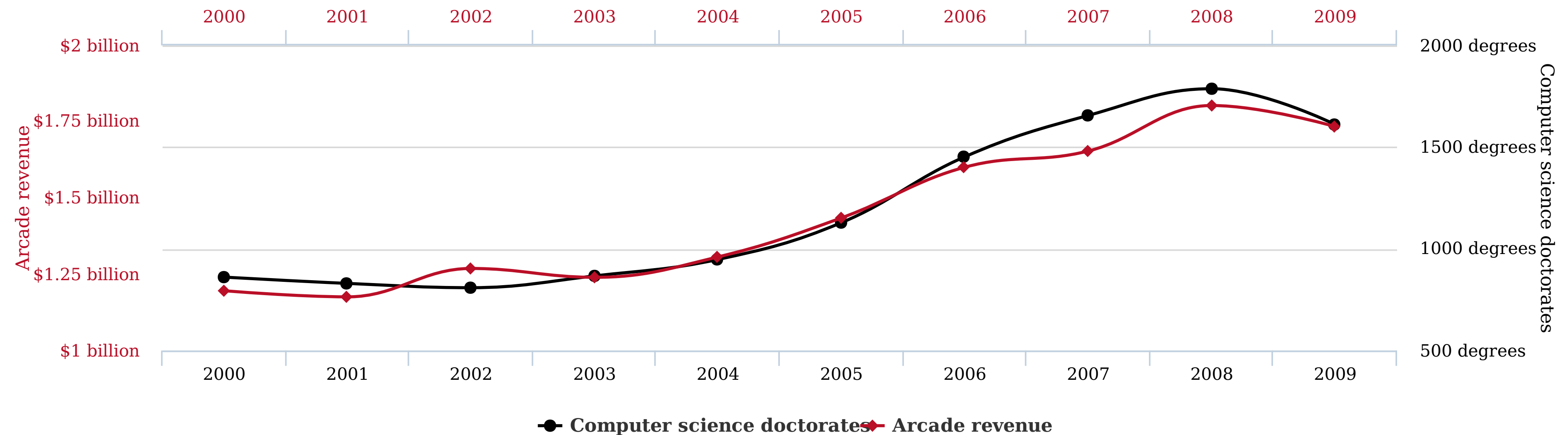


# Tensorflow

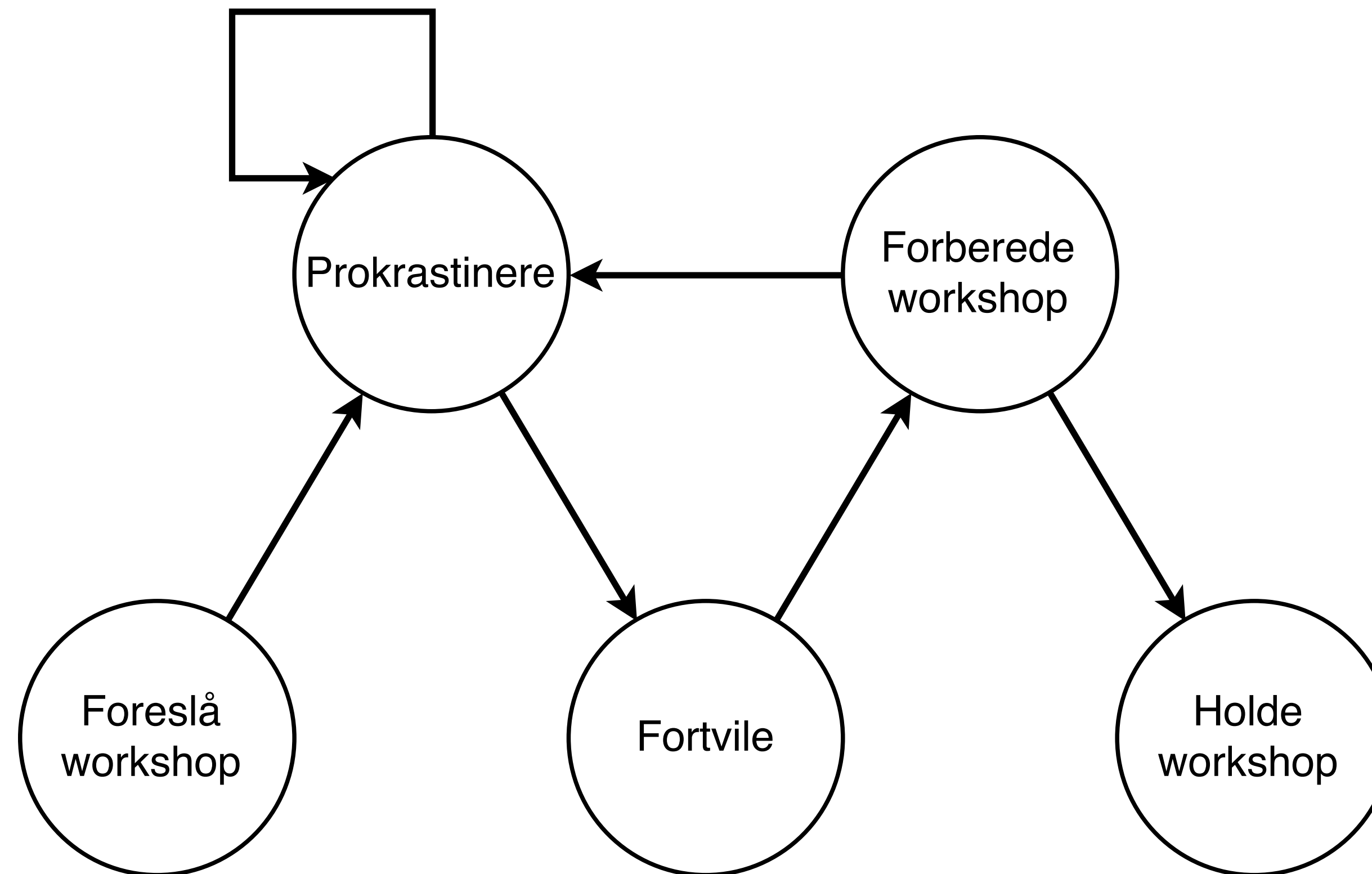
(egentlig TensorFlow, men altså...)

# Grafberegning

**Total revenue generated by arcades**  
correlates with  
**Computer science doctorates awarded in the US**



# Grafberegning



# Begreper

- Graph
- Session
- Tensor
- Constant, variable, placeholder
- Operation (op)

# Graph

- Konteksten innholdet du definerer havner i
- Sjelden behov for mer enn én graf

```
g = tf.Graph()  
with g.as_default():  
    # definer innhold
```

# Session

- Brukes til å kjøre en graf
- Kan evaluere hele eller subsett av grafen
- CPU eller GPU

```
sess = tf.Session()  
output = sess.run(<something_from_graph>)
```

# Tensor

- N-dimensjonalt array
- Input, output og state lagres i tensorer
- Tensorer kan brukes som de er, eller gjennom variabler og konstanter

# Constant

- Inneholder tensor med konstant verdi

```
data = [1, 2, 3]
constant = tf.constant(data)
out = sess.run(constant)
# out == [1, 2, 3]
```



# Variable

- Variabler har tensorer som lever i grafen
- Verdier huskes på tvers av sesjonskall
- Må initialiseres

```
const_init = tf.constant_initializer(7)
var = tf.get_variable(
    name='min_variabel',
    shape=[3],
    initializer=const_init)
sess.run(tf.initialize_all_variables())
three_sevens = sess.run(var)
```

# Placeholder

- Mekanisme for å gi input til grafen ved kjøring
- Fungerer som en peker til data utenfor grafen

```
input = tf.placeholder(  
    dtype=tf.int32,  
    shape=[3])  
feed_dict = { input: [1, 2, 3] }  
out = sess.run(input, feed_dict=feed_dict)  
# grafen evalueres med input == [1, 2, 3]  
# out er også her [1, 2, 3]
```

# Operation

- Representerer et stykke arbeid (som en funksjon)
- Kan returnere ingen eller flere tensorer

```
operation = tf.initialize_all_variables()  
sess.run(operation)
```

```
c = tf.constant([1, 2, 3])  
# operasjon for å summere c  
sum = tf.reduce_sum(c)  
out = sess.run(sum)  
# out == 6
```

# Aktivering

`tf.tanh` # Tangens, squasher til  $(-1,1)$

`tf.sigmoid` # Sigmoid, squasher til  $(0,1)$

`tf.nn.softmax` # Softmax sørger for at  
aktiveringene summerer til 1.0.  
Typisk anvendt for klassifisering.  
Omtales ofte som sannsynligheter.

`tf.relu` # Rectified linear unit,  
squasher nedre grense til 0.

# Optimizer

```
tf.train.GradientDescentOptimizer  
# implementert ved backpropagation.
```

```
tf.train.RMSPropOptimizer  
# bra for dype nett, f.eks 5+ skjulte lag
```