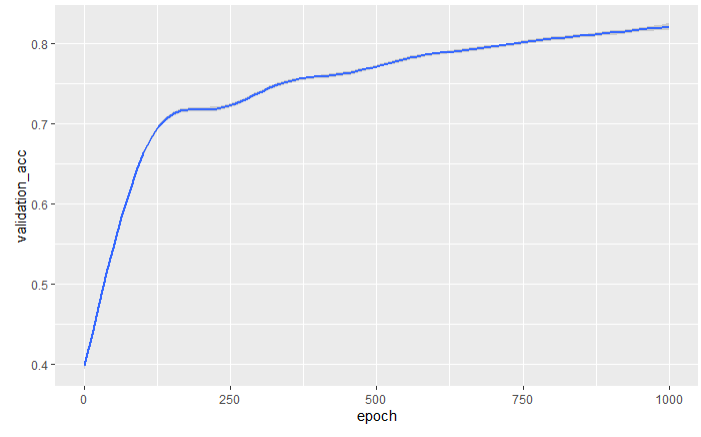
> head(max(average\_acc\_history$validation\_acc))

[1] 0.8342998

> head(eval)

loss categorical\_accuracy

0.8826595 0.6623530



build\_model <- function() {

model <- keras\_model\_sequential() %>%

#layer\_batch\_normalization(axis = -1L, input\_shape = dim(train\_data)[[2]]) %>%

layer\_dense(units = 128, activation = "relu", input\_shape = dim(train\_data)[[2]]) %>%

#layer\_dense(units = 128, activation = "relu") %>%

layer\_dropout(0.3) %>%

layer\_dense(units = 1024, activation = "relu") %>%

layer\_dropout(0.3) %>%

layer\_dense(units = 1024, activation = "relu") %>%

layer\_dropout(0.3) %>%

layer\_dense(units = 1024, activation = "relu") %>%

layer\_dropout(0.3) %>%

#layer\_dense(units = 128, activation = "relu") %>%

layer\_dense(units = 128, activation = "relu") %>%

layer\_dropout(0.3) %>%

layer\_dense(units = 8, activation = "softmax")

model %>% compile(

optimizer = optimizer\_sgd(learning\_rate = 0.1),

loss = "categorical\_crossentropy",

metrics = "categorical\_accuracy"

)

}

* Dropouts have a negative effect so far