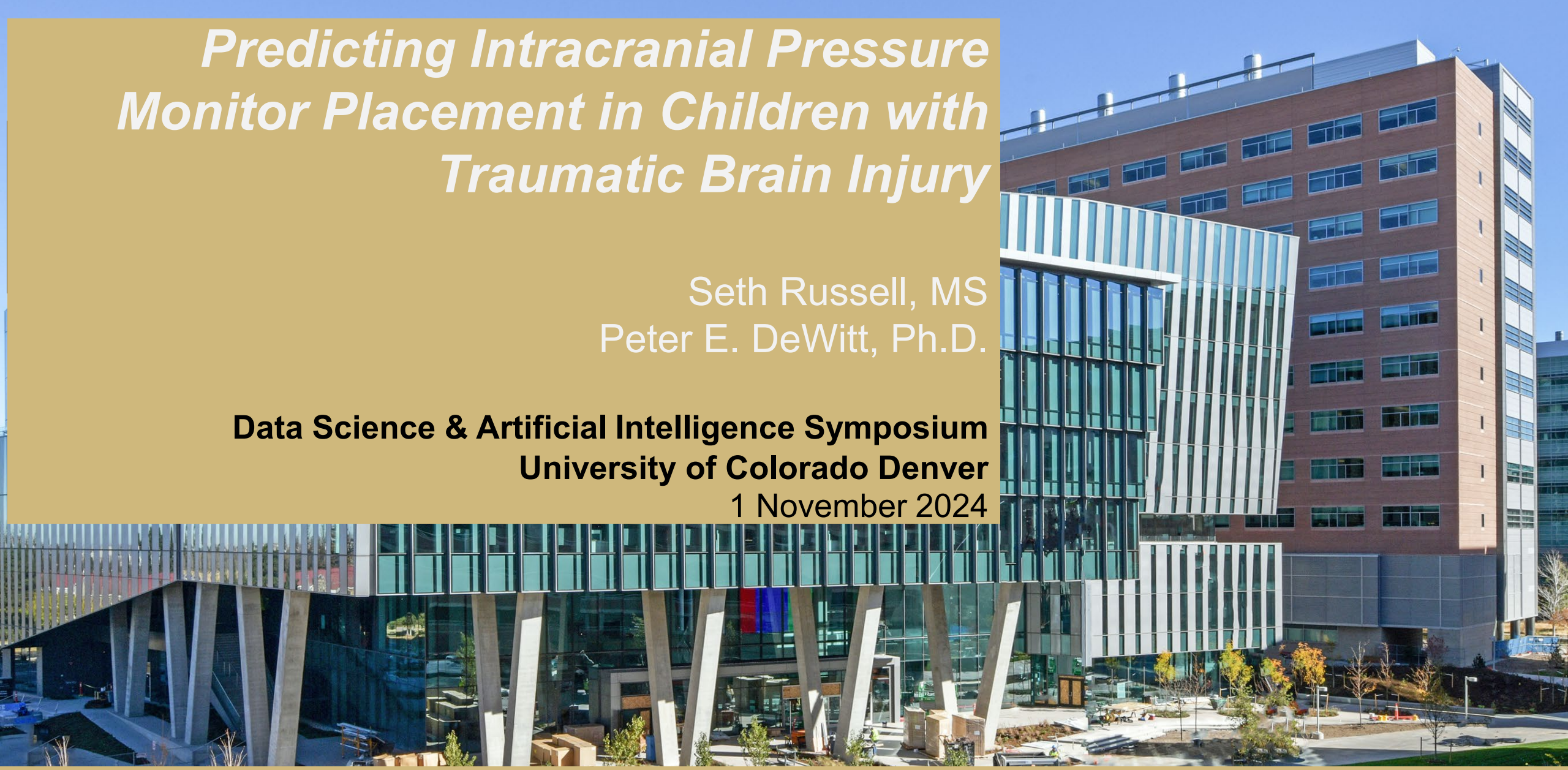
# *Predicting Intracranial Pressure Monitor Placement in Children with Traumatic Brain Injury*

Seth Russell, MS
Peter E. DeWitt, Ph.D.

**Data Science & Artificial Intelligence Symposium**
**University of Colorado Denver**
1 November 2024



Department of Biomedical Informatics
SCHOOL OF MEDICINE
UNIVERSITY OF COLORADO **ANSCHUTZ MEDICAL CAMPUS**

# Disclosures

This work was supported by:

- Mindsource Brain Injury Network of the Colorado Department of Human Services, PI Tell Bennett
- *Eunice Kennedy Shriver* National Institute for Child Health and Human Development, grant R03 HD094912, PI Tell Bennett

This work is currently under peer review for publication

Team of people contributed to this work:

Laura Helmkamp, MS; Kathryn Colborn PhD, MSPH; Charlotte Gray, MPH; Margaret Rebull, MA; Yamila L. Sierra, MPH; Rachel Greer, BA; Lexi Petruccelli, BA; Sara Shankman, CPNP-AC, DNP; Todd C. Hankinson, MD, MBA; Fuyong Xing, PhD; David J. Albers, PhD; and Tellen D. Bennett, MD, MS

https://github.com/magic-lantern/2024-dsai

# Motivation

Traumatic Brain Injury - A violent blow or jolt to the head or body that can cause temporary or permanent complications up to death. Swelling is a common serious complication.

Annually in the United States, traumatic brain injury causes:

- 2,200 pediatric deaths
- 35,000 pediatric hospitalizations

To monitor and track brain swelling, an invasive procedure is performed to install a device called an Intercranial Pressure Monitor (ICP) into the skull. Clinicians currently make decisions about ICP placement with traumatic brain injury without the benefit of an accurate clinical decision support tool

https://github.com/magic-lantern/2024-dsai

# Study

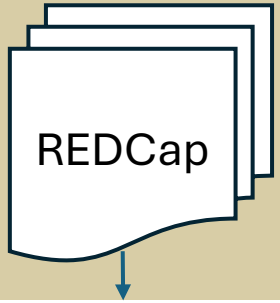
**Aim** - Develop and validate model(s) that predict placement of an ICP monitor. Models should update individual predictions as new information becomes available during an Intensive Care Unit (ICU) stay.

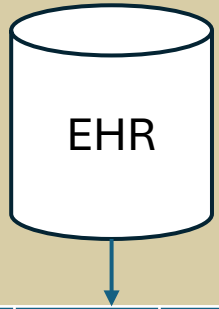
**Study**

- Prospective observational cohort study: September 2014 - January 2024
- 389 children with acute traumatic brain injury admitted to the ICU
- 138 (35%) of the 389 participants received ICP monitoring.
- Approved and granted a waiver of consent by the Colorado Multiple Institutional Review Board

https://github.com/magic-lantern/2024-dsai

# Data



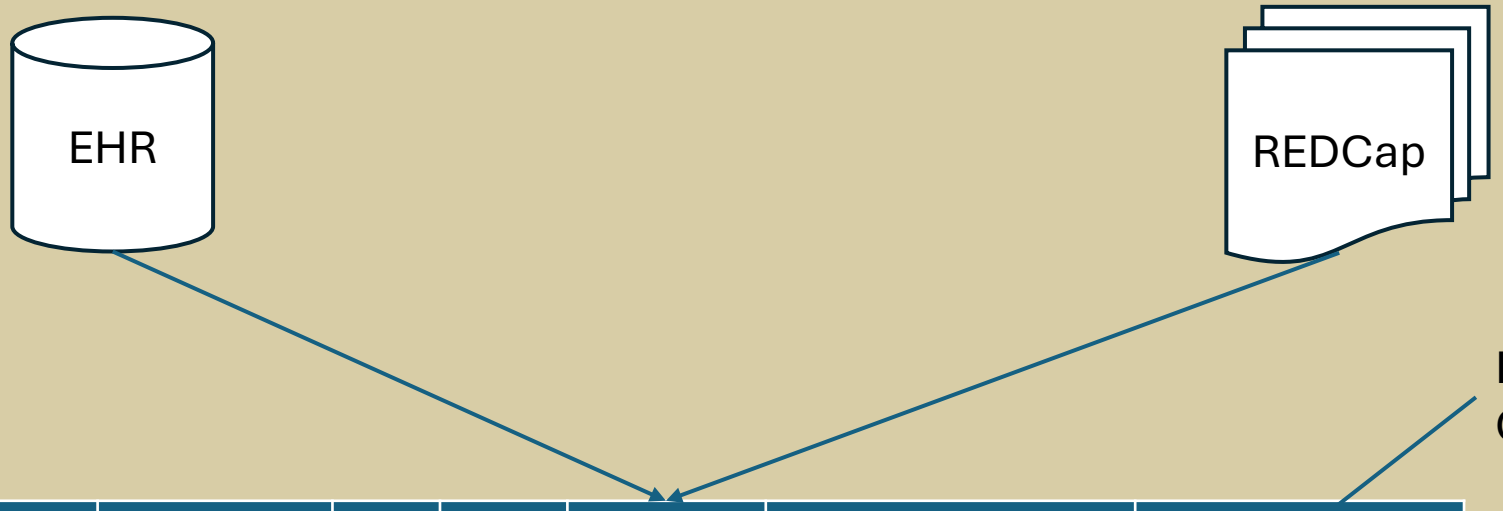

| Data Collection Instrument | Enrollment | Pre-hospital | ED | ICU | Acute Care | Rehab | Home |
|---|---|---|---|---|---|---|---|
| Screening and Enrollment | ● | | | | | | |
| Patient | ● | | | | | | |
| Injury | ● | | | | | | |
| Transitions | ● | | | | | | |
| ED | | | ● | | | | |
| ICU | | | | ● | | | |
| Sedating Medication | | ● | ● | ● | | | |
| Osmolar therapy | | ● | ● | ● | | | |
| Procedures | | ● | ● | ● | | | |
| Labs - sodium | | ● | ● | ● | | | |
| Labs - arterial CO2 | | ● | ● | ● | | | |
| Labs - venous CO2 | | ● | ● | ● | | | |
| Neurologic exam | | ● | ● | ● | ● | ● | ● |
| Initial CT | | | ● | | | | |
| Subsequent CT | | | | ● | | | |
| Functional Status Scale (FSS) | ● | | | ● | ● | ● | ● |
| Extended Glasgow Outcome Scale (GOSE Peds) | | | | | | | ● |
| Seizures | | | | ● | | | |
| ICP monitor | ● | | | | | | |

| record_id | vital | Units | Value | Datetime |
|---|---|---|---|---|
| 1 | pulse | bpm | 90 | 2027-01-01 13:01 |
| 1 | pulse oximetry | % | 98 | 2027-01-01 13:01 |
| 1 | respirations | | 24 | 2027-01-01 13:01 |
| 1 | pulse | bpm | 91 | 2027-01-01 14:01 |

In csv form, REDCap data is 276 columns wide

https://github.com/magic-lantern/2024-dsai

# Data




Last Observation Carry Forward

| record_id | time | ICP | age | lab_co2 | pulse | ... |
|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 13.1 | 40 | 120 | |
| 1 | 5 | 0 | 13.1 | 40 | 120 | |
| 1 | 10 | 0 | 13.1 | 40 | 121 | |
| 1 | 15 | 0 | 13.1 | 45 | 121 | |

Healthy default filled in for NULL

Plus more Data and Feature Engineering

~35 variables, 270k rows

# Data

**Exclude** CT extracted features and **exclude** Invasive BP

| record_id | time | ICP | age | lab_co2 | pulse | ... |
|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 13.1 | 40 | 120 | |
| 1 | 5 | 0 | 13.1 | 40 | 120 | |
| 1 | 10 | 0 | 13.1 | 40 | 121 | |
| 1 | 15 | 0 | 13.1 | 45 | 121 | |

**Include** CT extracted features and exclude Invasive BP

| record_id | time | ICP | age | lab_co2 | ct_fracture.factor | ct_ ... | ct_ ... | ... | ... |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 13.1 | 40 | 0 | | | | |
| 1 | 5 | 0 | 13.1 | 40 | 0 | | | | |
| 1 | 10 | 0 | 13.1 | 40 | 0 | | | | |
| 1 | 15 | 0 | 13.1 | 45 | 0 | | | | |

**Exclude** CT extracted features and **include** Invasive BP

| record_id | time | ICP | age | lab_co2 | inv_bp_available | ... |
|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 13.1 | 40 | 0 | |
| 1 | 5 | 0 | 13.1 | 40 | 0 | |
| 1 | 10 | 0 | 13.1 | 40 | 0 | |
| 1 | 15 | 0 | 13.1 | 45 | 1 | |

**Include** CT extracted features and **include** Invasive BP

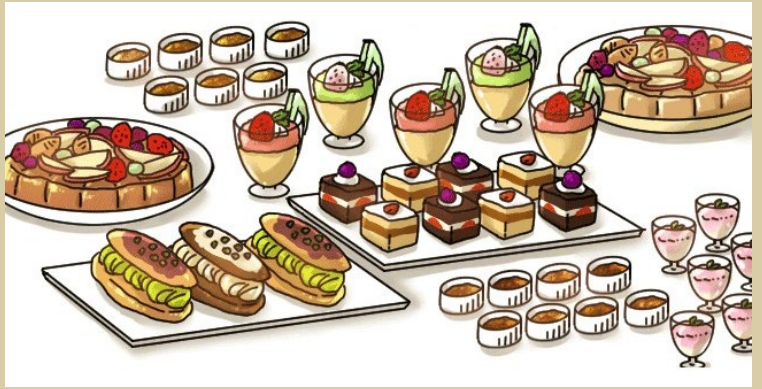
| record_id | time | ICP | age | ct_fracture.factor | ct_ ... | ct_ ... | inv_bp_available | ... |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 13.1 | 0 | | | 0 | |
| 1 | 5 | 0 | 13.1 | 0 | | | 0 | |
| 1 | 10 | 0 | 13.1 | 0 | | | 0 | |
| 1 | 15 | 0 | 13.1 | 0 | | | 1 | |

Additionally split data temporally into training and testing datasets

Moons KGM, Altman DG, Reitsma JB, *et al.* Transparent Reporting of a multivariable prediction model for Individual Prognosis or Diagnosis (TRIPOD): explanation and elaboration. *Ann Intern Med*. 2015;162:W1-73. doi: 10.7326/M14-0698

https://github.com/magic-lantern/2024-dsai

# Modeling



https://clipground.com/images/food-buffet-clipart-3.jpg

- Standard logistic regression
- Penalized logistic regression (lasso & ridge)
- Support vector machines
- Generalized estimating equations
- Generalized additive models
- Neural Network with Long Short-Term Memory layers

https://github.com/magic-lantern/2024-dsai

# Modeling – Regression p1

```r
#'
#' Define several formulae for the glm type models.  Same baseline inputs with
#' inclusion and exclusion of ct and invasive bp.
glm_formula <- list()

glm_formula$noct_noinvbp <-
  icpever ~
    timeint +
    age +
    female +
    inj_mvc +
    inj_abuse +
    # inj_fall +  # REFERENCE LEVEL for injury mechanism
    inj_other +
    ni_sbp +
    ni_dbp +
    etco2 +
    heartrate +
    spo2 +
    resprate +
    temp +
    total_score +
    motor_score +
  # pupils_bothreactive  + # REFERENCE LEVEL for pupils
    pupils_onefixed +
    pupils_bothfixed +
    pupils_unknown +
    lab_na +
    lab_co2 +
    lab_co2venous
```

# Modeling – Regression p2

```r
if (n_distinct(icp_data_3$noninv_ncnt_vit) > 1) {
  glm_formula$noct_noinvbp <- update(glm_formula$noct_noinvbp, . ~ . + noninv_ncnt_vit)
}

glm_formula$noct_invbp <- update(glm_formula$noct_noinvbp, . ~ . + inv_bp_available)

glm_formula$ct_noinvbp <-
  update(glm_formula$noct_noinvbp,
         . ~ . + ct_fracture.factor + ct_edema.factor + ct_shift.factor +
           ct_compression.factor + ct_intraparhem.factor + ct_subarachem.factor
         + ct_intravenhem.factor + ct_subduralhem.factor +
           ct_epiduralhem.factor)

glm_formula$ct_invbp <- update(glm_formula$ct_noinvbp, . ~ . + inv_bp_available)


#'
#' ### Training GLMs
#'
#' We will fit several models on the training data.
#'
#' #### Standard logistic regression models.
#'
#+ label = "standard glm training fit"
tic("Fitting GLM Models")
icpever_glm <-
  lapply(glm_formula,
         function(x) {
           glm(x,
               data = icp_data_3,
               subset = set == "Training",
               family = binomial())
         })
names(icpever_glm) <- paste0("glm_", names(icpever_glm))
```

# Modeling – LSTM p1

```
function(x_train,
         x_test,
         y_train,
         y_test,
         num_epochs = 10,
         batch_size = 10,
         validation_split = 0.2,
         dropout = 0.35,
         random_seed = RANDOM_SEED,
         tensorboard_label = 'run',
         opt = NULL,
         layers = c(128, 32, 16, 0),
         checkpoint_file = "model_checkpoint.keras") {
  # reset random seed and keras states so results are reproducible
  reset_state(random_seed)

  #### Network architecture definition
  seqmodel <- keras_model_sequential(input_shape = c(dim(x_train)[2], dim(x_train)[3]))
  # without input shape
  # building a model with no input shape results in exact same accuracy as with input shape
  # however, models without an `input_shape` passed to the first layer cannot reload their
  # optimizer state (cannot continue training)
  seqmodel %>%
    # note - keras allows specifying of some or none of the input_shape size
    # have to use list(NULL) as c(NULL) automatically discards the NULL entry
    # only specify predictors size for input_shape
    layer_masking(mask_value = 0) %>%
    layer_dropout(rate = dropout) %>%
    layer_dense(units = dim(x_train)[3], activation = "relu")
    # could try bidirectional LSTM with code like
```

# Modeling – LSTM p2

```
  # can specify layers as (layer1, layer2, ..., layern)
  # once a layer with 0 is encounted all subsequent values should be 0
  # e.g. (128, 64, 0, 0, 0) is ok
  #      (128, 0, 64, 0)    is invalid
  for (i in seq_along(layers)) {
    if (layers[i] > 0) {
      # if there is a next layer that is not empty, need to include return sequences for LSTM to work
      if((i + 1) <= length(layers) & layers[i + 1] > 0) {
        # print(paste("i:", i, "adding layer (", layers[i], ") with return_sequences"))
        seqmodel %>%
          layer_lstm(units = layers[i], return_sequences = TRUE)
          # tried `bias_initializer = "ones"` option, but made results worse given other
          # hyperparameters that already worked... Might need to start all over for good
          # results
      } else {
        # print(paste("i:", i, "adding layer (", layers[i], ") with NO return_sequences"))
        seqmodel %>%
          layer_lstm(units = layers[i])
      }
    }
  }
```

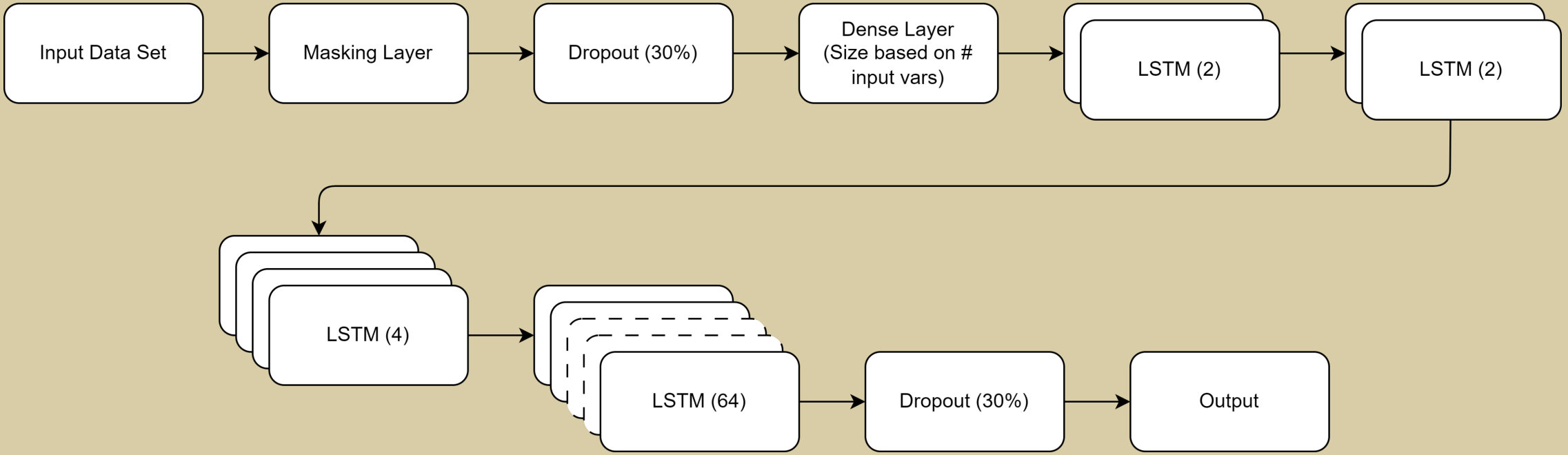
https://github.com/magic-lantern/2024-dsai

# Modeling – LSTM p3

```
54
55  seqmodel %>%
56    layer_dropout(rate = dropout) %>%
57    layer_dense(units = 1, activation = "sigmoid")
58  # with relu and 80 epochs validation accuracy 0.5492
59  # with sigmoid and 80 epochs validation accuracy 0.
60  # 88% end accuracy before adding dropout layer
61
62  if (is.null(opt)) {
63    seqmodel %>% compile(
64      loss = 'binary_crossentropy',
65      optimizer = 'rmsprop',
66      metrics = c('f1_score', 'binary_accuracy', tf$keras$metrics$Precision(), tf$keras$metrics$Recall()))
67  } else {
68    seqmodel %>% compile(
69      loss = 'binary_crossentropy',
70      optimizer = opt,
71      metrics = c('f1_score', 'binary_accuracy', tf$keras$metrics$Precision(), tf$keras$metrics$Recall()))
72  }
73
74  print('Model summary:')
75  summary(seqmodel)
76
77  checkpoint_file_path <- paste0(save_dir, "checkpoint/", checkpoint_file)
78  print(paste("Best model checkpoint saved to:", checkpoint_file_path))
79
```

https://github.com/magic-lantern/2024-dsai

# Modeling – LSTM p4




https://github.com/magic-lantern/2024-dsai

# Modeling – LSTM p5

```
  history <- seqmodel %>% fit(
    x_train,
    y_train,
    epochs = num_epochs,
    batch_size = batch_size,
    validation_split = validation_split,
    callbacks = list(
      callback_tensorboard(paste0(save_dir, "tensorboard/", tensorboard_label)),
      callback_reduce_lr_on_plateau(monitor = "val_loss",
                                    factor = 0.5,
                                    patience = 20,
                                    verbose = 1,
                                    min_lr = 0.0000001),
      callback_early_stopping(monitor = "val_loss",
                              min_delta = 0.0001,
                              patience = 40,
                              verbose = 1,
                              mode = c("min")),
      # since we have early stopping, this shouldn't be necessary
      # also checkpoint can only be every epoch, so will greatly increase disk usage
      callback_model_checkpoint(checkpoint_file_path,
                                monitor = "val_loss",
                                save_best_only = TRUE,
                                save_weights_only = FALSE,
                                mode = "min",
                                save_freq = "epoch")
    )
    # , shuffle = FALSE # this option results in >2x more epochs to get to same accuracy
    # include this to adjust model weights based on class
    #class_weight = list("0"=1,"1"=length(which(y_train==0)) / length(which(y_train==1))),
  )
  return(list('model' = seqmodel, 'history' = history))
}
```

# Modeling LSTM Hyperparameters

```
num_simultaneous_jobs <- 7
EPOCH_RUN_TIME <- 40 # time in seconds based on single simultaneous job
job_file <- "jul10best.txt"
k_folds <- 5 # normally use 5 when finding best hyperparameters
# at least with other chosen hyperparameters, 200 epochs gives the best result
# doing early stopping if no improvements, so will just use 500 for everything
epochs <- c(500)
batch_size <- c(1024) # seems to have little effect on accuracy; big difference
learn_rate <- c(0.1, 0.05, 0.01, 0.005)
validation_split <- c(0.2, 0.3, 0.4)
dropout <- c(0.05, 0.1, 0.2, 0.3, 0.4)
run_final <- TRUE
# which optimizers to use? see:
# http://cs231n.github.io/neural-networks-3/
# https://ruder.io/optimizing-gradient-descent/index.html
optimizer <- c(
                  "rmsprop=optimizer_rmsprop",
                  "adam=optimizer_adam",
                  "sgd=optimizer_sgd",
                  "adadelta=optimizer_adadelta"
                  )
predict_advance_times <- c(0)
time_block_size <- 5
icpever_pred <- TRUE

# create grid of network architecture options from these variables
num_layers <- seq(4)
layer_size <- c(2, 4, 8, 16, 32, 48, 64)

random_seed <- c(42, 634906796, 1977035194, 1699415226, 1673646895, 1393266542,
                 1083032605, 136228259, 1361975978, 1627064900, 43586684)

# primarily for final run, want to compare model against restricted input model
exclude_vars <- c('"inv","ct"', '"ct"', '"inv"', "")
```

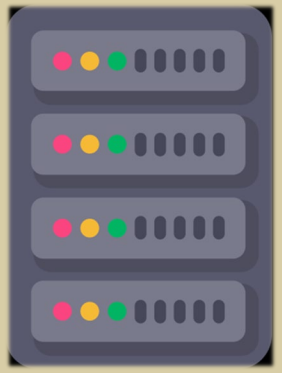
# Modeling - LSTM hyperparm

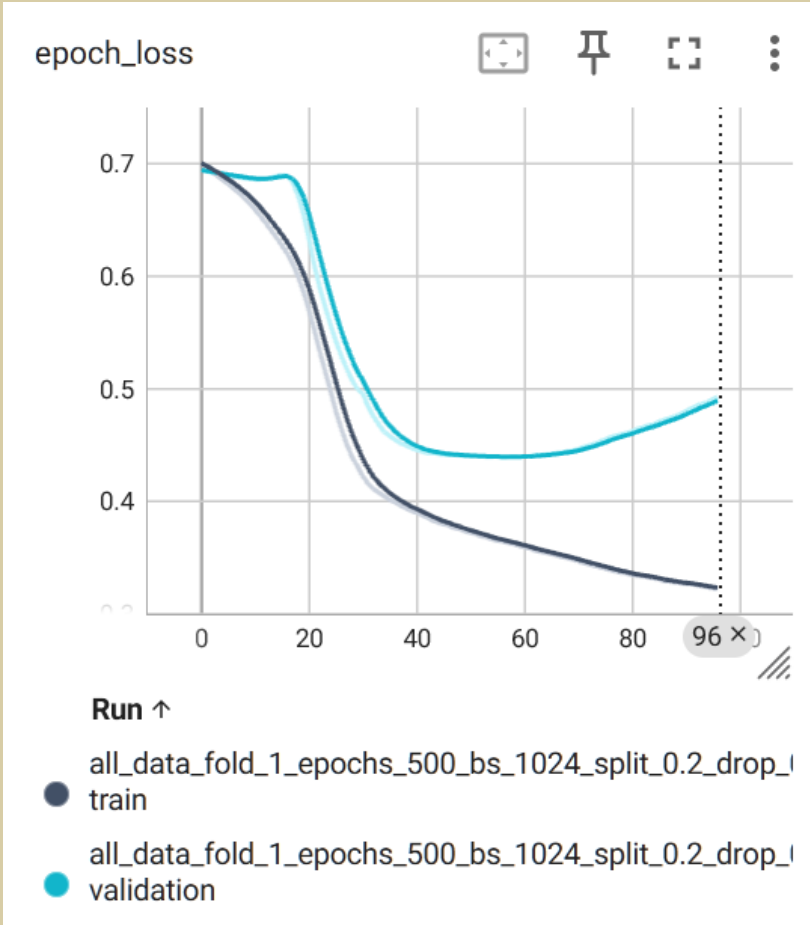
```
base_g <- expand.grid(k_folds = k_folds,
                      epochs = epochs,
                      batch_size = batch_size,
                      learn_rate = learn_rate,
                      validation_split = validation_split,
                      dropout = dropout,
                      random_seed = random_seed,
                      optimizer = optimizer,
                      predict_advance_times = predict_advance_times,
                      run_final = run_final,
                      time_block_size = time_block_size,
                      icpever_pred = icpever_pred,
                      exclude_vars = exclude_vars)

if (!is.na(layers)) {
  base_g$layers <- layers
  g <- as.data.table(base_g)
} else {
  layer_g <- expand.grid(num_layers = num_layers,
                         layer1_size = layer_size,
                         layer2_size = layer_size,
                         layer3_size = layer_size,
                         layer4_size = layer_size)

  # not sure how to get expand.grid to do what I want, so cleanup and remove extra stuff
  g <- layer_g %>%
    dplyr::mutate(layer2_size = ifelse(num_layers <= 1, NA, layer2_size)) %>%
    dplyr::mutate(layer3_size = ifelse(num_layers <= 2, NA, layer3_size)) %>%
    dplyr::mutate(layer4_size = ifelse(num_layers <= 3, NA, layer4_size)) %>%
    tidyr::unite("layers", layer1_size, layer2_size, layer3_size, layer4_size, sep = ",", na.rm = TRUE) %>%
    dplyr::select(-num_layers) %>%
    unique

  gdt <- as.data.table(g)
  base_gdt <- as.data.table(base_g)
  g <- unique(CJ.dt_2(gdt, base_gdt))
}
```

# Modeling – LSTM training

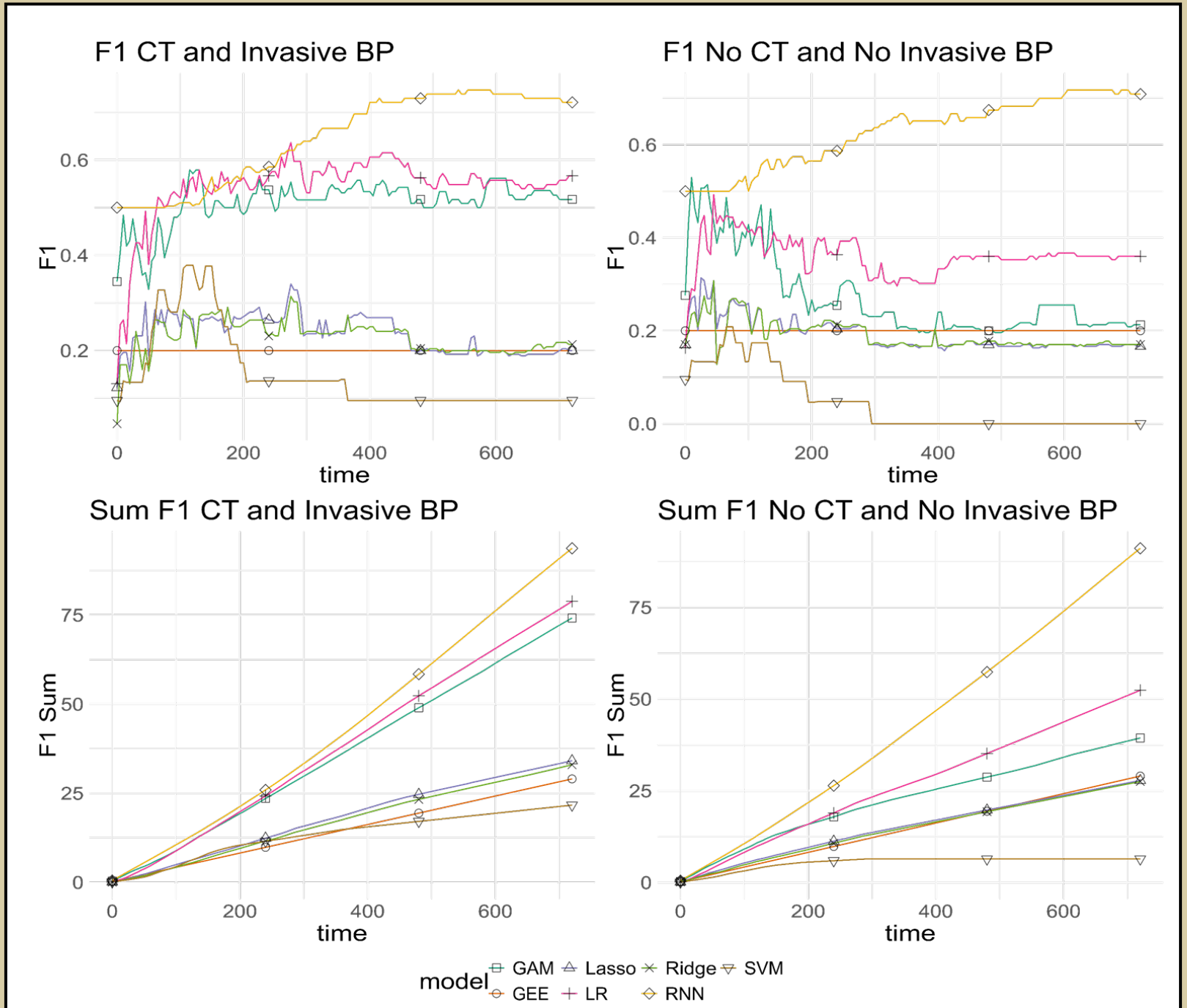
```
cd ~/aim2data_long_job
# run these jobs on GPU0
CUDA_VISIBLE_DEVICES=0  parallel --jobs 4  --delay 120 --timeout=14400 --resume-failed --bar --joblog rnd_gpu0.log < rnd_gpu0_jobs.txt
# Run these jobs on GPU1
CUDA_VISIBLE_DEVICES=1  parallel --jobs 4  --delay 120 --timeout=14400 --resume-failed --bar --joblog rnd_gpu1.log < rnd_gpu1_jobs.txt
# run these jobs on CPU
CUDA_VISIBLE_DEVICES=-1 parallel --jobs 28 --delay 120 --timeout=86400 --memfree 150G --resume-failed --bar --joblog rnd_cpu.log  < rnd_cpu_jobs.txt
```

(4) Intel Xeon Gold 6416H 'Sapphire Rapids-SP' 2.2 GHz 18-core CPU
(32) 64GB DDR5 4800 MHz ECC/Registered Memory
(2) NVIDIA "Ampere" A100 PCI-E+NVLink 80GB
(4) 7.68TB Intel SSD D3-S4520 2.5" SATA 6Gbps
(1) 15.36TB Micron 9400 PRO 2.5" U.3 NVMe SSD



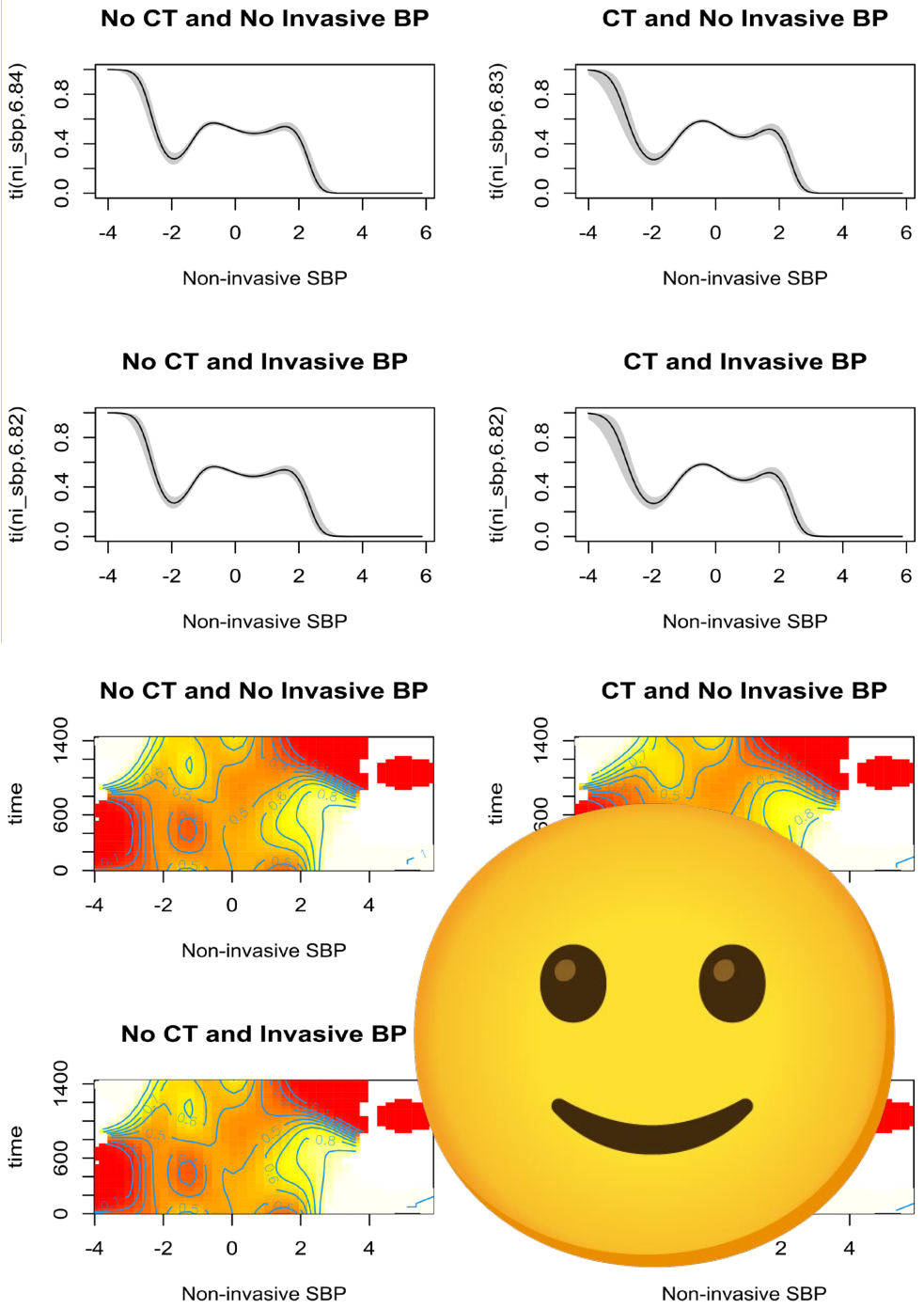


# Results


F1 CT and Invasive BP
F1 No CT and No Invasive BP
Sum F1 CT and Invasive BP
Sum F1 No CT and No Invasive BP
F1
F1 Sum
time
0
200
400
600
0.2
0.4
0.6
0.0
25
50
75
model
GAM
Lasso
Ridge
SVM
GEE
LR
RNN

# Interpretability - Linear

| Variable | LR | Lasso | Ridge | Absolute Rank |
|---|---|---|---|---|
| CT - Intraparenchymal Hemorrhage | 0.88 | 0.864 | 0.676 | 2 |
| Injury Mechanism - Known or Suspected Abuse | 0.903 | 0.877 | 0.559 | 2 |
| CT - Cerebral edema | 0.742 | 0.734 | 0.604 | 4 |
| CT - Midline shift | 0.335 | 0.327 | 0.288 | 10 |
| Pupils - Both Fixed | 0.295 | 0.292 | 0.266 | 11 |
| Age | 0.29 | 0.284 | 0.191 | 13 |
| Non-invasive DBP | 0.279 | 0.277 | 0.238 | 13 |
| CT - Skull fracture | 0.287 | 0.273 | 0.174 | 16 |
| Pupils - Unknown | 0.249 | 0.243 | 0.201 | 17 |
| Injury Mechanism - Motor Vehicle Crash | 0.274 | 0.255 | 0.104 | 19 |
| Arterial pCO2 | 0.193 | 0.187 | 0.195 | 19 |
| Injury Mechanism - Other | 0.258 | 0.241 | 0.111 | 21 |
| Invasive BP available | 0.108 | 0.105 | 0.095 | 24 |
| Female | 0.105 | 0.098 | 0.072 | 26 |
| Oxygen Saturation (SpO2) | 0.097 | 0.093 | 0.084 | 26 |
| Non-invasive SBP | 0.075 | 0.071 | 0.079 | 28 |
| CT - Basilar cistern compression | 0.082 | 0.075 | 0.052 | 29 |
| Venous pCO2 | 0.051 | 0.049 | 0.061 | 31 |
| Number of Vitals Updated | 0.028 | 0.026 | 0.038 | 32 |
| Pupils - One Fixed | 0.029 | 0.026 | 0.007 | 33 |
| Intercept | -0.938 | -2.448 | -2.251 | 1 |
| CT - Intraventricular Hemorrhage | -0.62 | -0.614 | -0.528 | 5 |
| CT - Subdural Hematoma | -0.478 | -0.467 | -0.336 | 6 |
| CT - Epidural Hematoma | -0.432 | -0.419 | -0.337 | 7 |
| GCS-Motor | -0.378 | -0.382 | -0.346 | 8 |
| Heart Rate | -0.358 | -0.355 | -0.295 | 9 |
| Time Interval | -0.002 | -0.879 | -0.717 | 12 |
| CT - Subarachnoid Hemorrhage | -0.292 | -0.274 | -0.154 | 15 |
| Respiratory Rate | -0.25 | -0.245 | -0.177 | 18 |
| End-tidal CO2 | -0.147 | -0.145 | -0.144 | 22 |
| Temperature | -0.128 | -0.123 | -0.107 | 23 |
| Sodium Level | -0.114 | -0.106 | -0.067 | 25 |
| GCS-Total | -0.033 | -0.031 | -0.092 | 29 |



# Interpretability - LSTM

# Questions?

Slides available at:

https://github.com/magic-lantern/2024-dsai