

SAS[®] GLOBAL FORUM 2019

USERS PROGRAM

APRIL 28 - MAY 1, 2019 | DALLAS, TX



INTRODUCTION

- The Acute Physiology and Chronic Health Evaluation (APACHE) II is a severity of disease classification system used in intensive care units (ICU)
- Severity scores are used in predicting patient outcomes, grouping patients for clinical trials, and comparing quality of care
- Not all electronic medical records support the automatic calculation of severity scores, leaving physicians and researchers to calculate the APACHE II score by hand or using online calculators
- We developed a SAS macro using real ICU admissions data to calculate APACHE II scores

BACKGROUND

- Revision of the prototype APACHE system attempting to simplify but maintain a statistically accurate patient classification assessment
- Developed in 1985 using a database of North American ICU patients
- Based on the hypothesis that acute disease severity can be measured by quantifying the degree of abnormality of clinically relevant physiologic variables
- The score is made up of three components:
 - acute physiology score (APS): weighted on a 0-4 point scale
 - age: scored higher for older patients
 - chronic health conditions: scored based on condition and operative metrics

Table 1: The Acute Physiology and Chronic Health Evaluation II Severity Scale

Acute Physiology Score (APS)

Physiologic Variable	High Abnormal Range					Low Abnormal Range			
	+4	+3	+2	+1	0	+1	+2	+3	+4
Temperature – rectal (°C)	≥41	39 to 40.9		38.5 to 38.9	36 to 38.4	34 to 35.9	32 to 33.9	30 to 31.9	≤29.9
Mean Arterial Pressure (mm Hg)	≥160	130 to 159	110 to 129		70 to 109		50 to 69		≤49
Heart Rate (ventricular response)	≥180	140 to 179	110 to 139		70 to 109		55 to 69	40 to 54	≤39
Respiratory Rate (non-ventilated or ventilated)	≥50	35 to 49		25 to 34	12 to 24	10 to 11	6 to 9		≤5
Oxygenation: A-aDO ₂ or PaO ₂ (mm Hg)	≥500	350 to 499	200 to 349		<200				
a. FIO ₂ ≥0.5 record A-aDO ₂					PO ₂ >70	PO ₂ 61 to 70		PO ₂ 55 to 60	PO ₂ <55
b. FIO ₂ <0.5 record PaO ₂									
Arterial pH (preferred)	≥7.7	7.6 to 7.69		7.5 to 7.59	7.33 to 7.49		7.25 to 7.32	7.15 to 7.24	<7.15
Serum HCO ₃ (venous mEq/l) (not preferred but may use if no ABGs)	≥52	41 to 51.9		32 to 40.9	22 to 31.9		18 to 21.9	15 to 17.9	<15
Serum Sodium (mEq/l)	≥180	160 to 179	155 to 159	150 to 154	130 to 149		120 to 129	111 to 119	≤110
Serum Potassium (mEq/l)	≥7	6 to 6.9		5.5 to 5.9	3.5 to 5.4	3 to 3.4	2.5 to 2.9		<2.5
Serum Creatinine (mg/dl)	≥3.5	2 to 3.4	1.5 to 1.9		0.6 to 1.4		<0.6		
Double point score for acute renal failure									
Hematocrit (%)	≥60		50 to 59.9	46 to 49.9	30 to 45.9		20 to 29.9		<20
White Blood Count (total/mm ³) (in 1000s)	≥40		20 to 39.9	15 to 19.9	3 to 14.9		1 to 2.9		<1

Age Points

≤ 44	+0
45-54	+2
55-64	+3
65-74	+5
≥75	+6

Chronic Health Conditions

Liver	If the patient has a prior organ system insufficiency or is immunocompromised: +5 for non-operative or emergency postoperative patients +2 for elective postoperative patients
Cardiovascular	
Respiratory	
Renal	
Immunocompromised	

Total APACHE II Score = APS + Age + Chronic Health Conditions

APACHE II SCORE

Acute Physiology Score (APS)

- Points are assigned based on the most extreme value seen within 24 hours of ICU admission of 12 physiologic variables

Age

- Chronological age is well known risk factor for death from acute illness regardless of disease severity

Chronic Health Conditions

- Patients with a history of a chronic health condition have decreased probability of survival during acute illness

INTERPRETATION

- Scores can range from 0-71, where an increase in score is correlated with an increase in death rate

Table 2: Death Rates

Score	Death Rate %
0-4	4
5-9	8
10-14	15
15-19	25
20-24	40
25-29	55
30-34	75
> 34	85

APACHE II MACRO

- Contains one data step that uses supplied values to generate the score
- Uses series of if-then statements to assign points based on the input values and to determine what is the most extreme value
- All variable inputs must be numeric
- For a full list of macro inputs and the full macro, see paper appendix

EXAMPLES

```
*temperature;
if (&tempmin >= 41) or (&tempmin <= 29.9) then tempmin_pts = 4;
else if (&tempmin >= 39 and &tempmin <= 40.9) or
    (&tempmin >= 30 and &tempmin <= 31.9) then tempmin_pts = 3;
else if (&tempmin >= 32 and &tempmin <= 33.9) then tempmin_pts = 2;
else if (&tempmin >= 38.5 and &tempmin <= 38.9) or
    (&tempmin >= 34 and &tempmin <= 35.9) then tempmin_pts = 1;
else if (&tempmin >= 36 and &tempmin <= 38.4) then tempmin_pts = 0;

if (&tempmax >= 41) or (&tempmax <= 29.9) then tempmax_pts = 4;
else if (&tempmax >= 39 and &tempmax <= 40.9) or
    (&tempmax >= 30 and &tempmax <= 31.9) then tempmax_pts = 3;
else if (&tempmax >= 32 and &tempmax <= 33.9) then tempmax_pts = 2;
else if (&tempmax >= 38.5 and &tempmax <= 38.9) or
    (&tempmax >= 34 and &tempmax <= 35.9) then tempmax_pts = 1;
else if (&tempmax >= 36 and &tempmax <= 38.4) then tempmax_pts = 0;

if tempmin_pts >= tempmax_pts then apache_score = apache_score +
tempmin_pts;
else if tempmin_pts < tempmax_pts then apache_score = apache_score
+ tempmax_pts;
```

Table 3: Temperature Points

obs	tempmin	tempmax	tempmin_pts	tempmax_pts	To be added in score:
1	32.5	36.6	2	0	2
2	35.4	40.0	1	3	3
3	36.0	37.6	0	0	0
4	38.1	39.6	0	3	3
5	36.5	37.6	0	0	0

- Other APS variables that require minimum and maximum values assign points with similar blocks for code as shown above for temperature

```
*oxygenation;
if &fio2max >= 0.5 then do;
    aado2 = (699.3*(&fio2max)) - (&po2min / 0.8) - (&pco2);
    if aado2 >= 500 then oxygen_pts = 4;
    else if aado2 >= 350 and aado2 <= 499 then oxygen_pts = 3;
    else if aado2 >= 200 and aado2 <= 349 then oxygen_pts = 2;
    else if aado2 < 200 then oxygen_pts = 0;
end;

else do;
    if &po2min > 70 then oxygen_pts = 0;
    else if &po2min >= 61 and &po2min <= 70 then oxygen_pts = 1;
    else if &po2min >= 55 and &po2min <= 60 then oxygen_pts = 3;
    else if &po2min < 55 then oxygen_pts = 4;
end;

apache_score = apache_score + oxygen_pts;
```

Table 4: Oxygenation Points

obs	oxyfio2max	po2min	pco2	aado2	oxygen_pts	To be added in score:
2	0.8	113	26	392.19	3	3
11	0.8	159	24	336.69	2	2
20	0.4	96	38	.	0	0
21	0.4	71	31	.	0	0
22	0.4	116	49	.	0	0

```
*glasgow coma score;
gcs_pts = 15 - &gcsmin;

apache_score = apache_score + gcs_pts;

*age points;
if &age < 45 then age_pts = 0;
else if &age >= 45 and &age < 55 then age_pts = 2;
else if &age >= 55 and &age < 65 then age_pts = 3;
else if &age >= 65 and &age < 75 then age_pts = 5;
else if &age >= 75 then age_pts = 6;

apache_score = apache_score + age_pts;

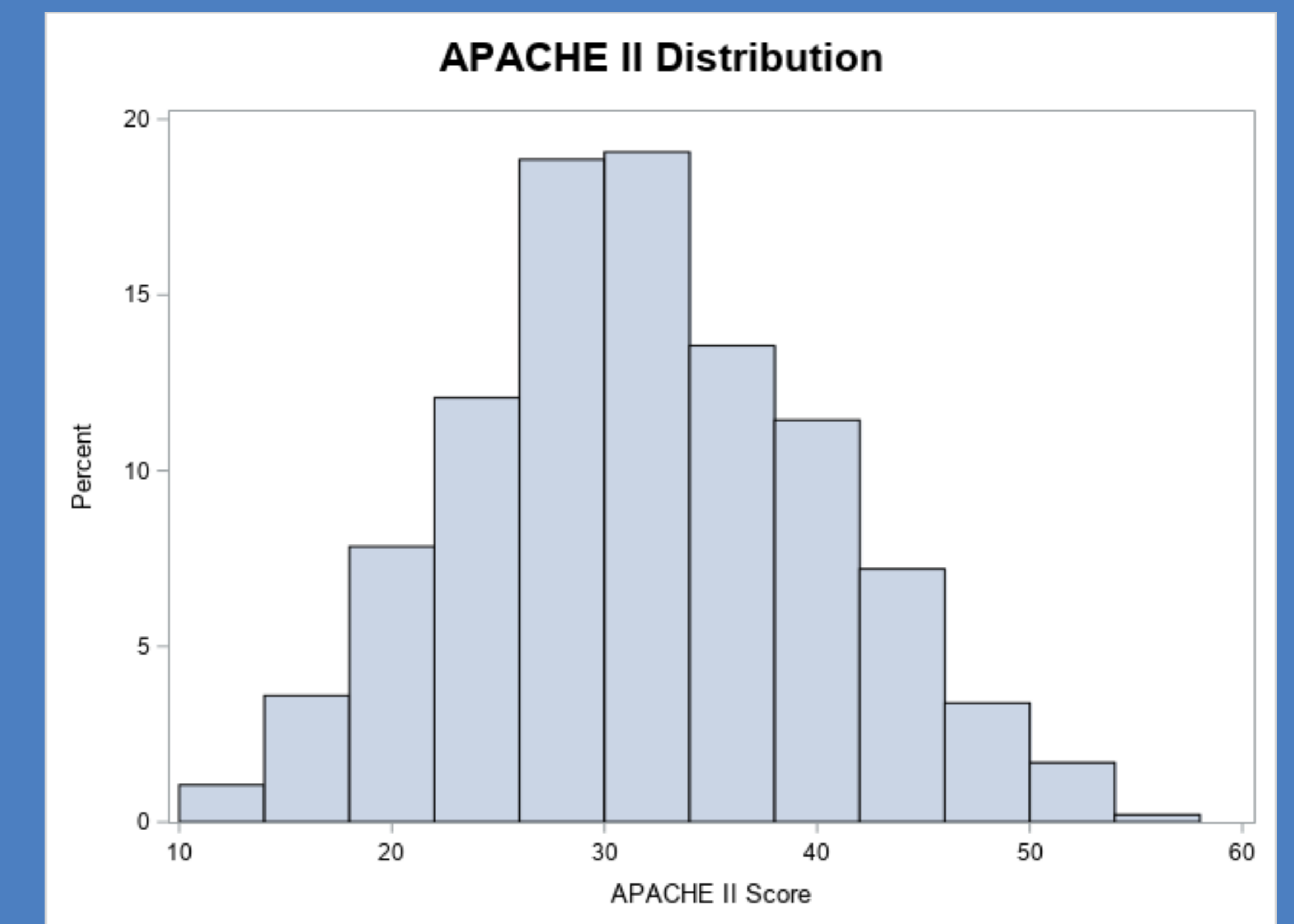
*chronic health conditions;
if &chc = 0 then chc_pts = 0;
else if &chc = 1 and &surg in(0 1) then chc_pts = 5;
else if &chc = 1 and &surg = 2 then chc_pts = 2;

apache_score = apache_score + + chc_pts;
```

Table 5: GCS, Age, and Chronic Health Condition Points

obs	gcsmin	gcs_pts	age	age_pts	chc	surg	chc_pts	To be added in score:
1	3	12	83	6	1	1	5	23
2	3	12	66	5	0	.	0	17
3	3	12	84	6	0	.	0	18
4	7	8	62	3	0	.	0	11
5	9	6	81	6	1	1	5	12

- After invoking the whole macro, points attributed to each variable and the total APACHE II score will be added as variables to your dataset
- The histogram below shows the distribution of APACHE II scores within our sample



REFERENCES

Knaus, W.A., Draper, E.A., Wagner, D.P. & Zimmerman, J.E. (1985). APACHE II: a severity of disease classification system. Critical Care Medicine, 13, 818-829.

#SASGF

SAS[®]
GLOBAL
FORUM
2019

APRIL 28 - MAY 1, 2019 | DALLAS, TX

Kay Bailey Hutchison Convention Center

Nothing is More Powerful than the Apache Helicopter, Except the APACHE II Macro

Margaret M. Kline, Grand Valley State University and Spectrum Health Office of Research; Jessica L. Parker, Spectrum Health Office of Research

ABSTRACT

The Acute Physiology and Chronic Health Evaluation (APACHE) II classification system is commonly used in intensive care units (ICU) to classify disease severity and predict hospital mortality. Disease severity scales in the ICU are a necessary component to assist in predicting patient outcomes, comparing quality of care, and stratifying patients for clinical trials. The APACHE II score is calculated from patient demographics and physiologic variables measured in the patient's first 24 hours following ICU admission. The score comprises three components: 1) an acute physiology score (APS); 2) age; and 3) chronic health conditions. Although this information is accessible in electronic medical records (EMR), some systems do not have a way to automate the calculation of the APACHE II score, leaving physicians to calculate it by hand, which decreases patient care time and increases calculation error rates. This paper shows how to calculate the APACHE II score based on information from EMRs using our macro.

INTRODUCTION

Disease severity scores aide physicians in clinical decision making, evaluating treatment effectiveness, optimizing hospital resources, and development of treatment standards. Physicians also use these scores to stratify patient populations for clinical trials and publications. The scoring systems are usually made up of two parts—a raw score which corresponds to disease severity and a probability model that estimates the probability of hospital death. For intensive care unit (ICU) patients, their outcome is dependent on their first day in the ICU (Rapsang & Shyam, 2014). Many of the scoring systems rely on measurements taken within the patient's first 24 hours in the ICU. APACHE II, MODS, SOFA, and LODS are some of the commonly used ICU scoring systems for adults.

As a result of the Health Information Technology for Economic and Clinical Health (HITECH) Act of 2009 aiming to stimulate electronic medical records (EMR) adoption by hospitals and medical practices, 96% of acute care hospitals in the United States have an EMR that meets requirements set by the Department of Health and Human Services as of 2017 (Office of the National Coordinator for Health Information Technology). Along with the movement to electronic records, there has been a push for practicing evidence-based medicine using the information gained from electronically storing clinical data. Integration of useful evidence-based clinical score models into the EMR with automated calculation based on real-time data is a logical step towards continuing to improve patient care (Aarke, Dziadzko & Herasevich, 2017). Unfortunately, most EMRs do not support the capability of automating these scores. Physicians are left to calculate scores by hand or with online calculators, both time consuming and error prone human processes that consequently prohibits the use of severity scores and impedes data collection in clinical research. This paper explains how to use data extracted from the EMR to calculate an APACHE II score using a SAS® macro.

APACHE II SEVERITY OF DISEASE CLASSIFICATION SYSTEM

The Acute Physiology and Chronic Health Evaluation (APACHE) II is a severity of disease classification system that was developed in 1985 using a database of North American ICU

patients. APACHE II is a revision of the prototype system, APACHE, and is the result of simplification of the original system to provide a more clinically useful yet statistically accurate patient classification assessment. It uses a point score based on 12 physiologic measurements, age, and chronic health status to provide one summary measure of disease severity.

ACUTE PHYSIOLOGY SCORE (APS)

The first component of the APACHE II score is the acute physiology score (APS). The original APACHE system was based on the hypothesis that severity of acute disease can be measured by quantifying the degree of abnormality of clinically relevant physiologic variables. APACHE II is still based on this hypothesis, but the number of variables measured has been reduced to 12 from the original 34. The 12 necessary physiological variables are: temperature, mean arterial pressure, heart rate, respiratory rate, oxygenation values, arterial pH, serum sodium, serum potassium, serum creatinine, hematocrit, white blood cell count, and Glasgow Coma Score (GCS). All the measurements must have been taken within the patient’s first 24 hours of ICU admission; a time frame which ensures all relevant physiologic variables are available and clinically accurate.

The APS is based on a weighting system on a scale from 0 – 4. Each physiologic variable has different cutoffs of values which are assigned different weights. Table 1 shows the points associated with the different cutoffs for temperature. For a full table of all 12 variables and their cutoff points, see Table 4 in the Appendix. Points are assigned based on the most extreme value seen within the 24-hour window. For example, a patient’s APACHE II score would be elevated 3 points if their temperature was measured at 40.9°C at some point within 24 hours of ICU admission, and all other temperatures taken had been between the normal range (36 to 38.4°C).

	+4	+3	+2	+1	0	+1	+2	+3	+4
Temperature – rectal (°C)	≥41	39 to 40.9		38.5 to 38.9	36 to 38.4	34 to 35.9	32 to 33.9	30 to 31.9	≤29.9

Table 1 Temperature Points

AGE AND CHRONIC HEALTH POINTS

Because age and chronic health conditions reflect physiologic reserve, the second and third components of the APACHE II score are points related to the age and chronic health problems of the patient. Chronological age is a well-known risk factor for death from acute illness, regardless of severity of disease, and patients with a history of severe organ system insufficiency or who are immunocompromised have decreased probability of survival during acute illnesses (Knaus et al., 1985). Table 2 shows the weights assigned based on the patient’s age.

Age (years)	Points
≤ 44	0
45-55	2
55-64	3
65-74	5
≥ 75	6

Table 2 Age Points

The chronic health condition must have been evident before hospital admission and conform to a certain set of criteria for these additional points to be added to the total APACHE II score. The following criteria as defined by Knaus et al. specify the five possible prior chronic health conditions:

- Liver- proven cirrhosis and documented portal hypertension, episodes of upper GI bleeding attributed to portal hypertension, or prior episodes of hepatic failure, encephalopathy and/or coma
- Cardiovascular- New York Heart Classification of Class IV
- Respiratory- chronic restrictive, obstructive, or vascular disease resulting in severe exercise restriction, such as being unable to climb stairs or perform household duties, or documented chronic hypoxia, hypercapnia, secondary polycythemia, severe pulmonary hypertension (> 40mmHg), or respirator dependency
- Renal- receiving chronic dialysis
- Immunocompromised- receiving therapy that suppresses resistance to infection, such as immunosuppression, chemotherapy, radiation, or long term or recent high dose steroids, or has a disease that is sufficiently advanced to suppress resistance to infection, e.g. leukemia, lymphoma, AIDS

Patients with any of the above conditions receive additional points to their score. During the validation of the APACHE II score, it was found that non-operative and emergency surgery admissions had a higher risk of death from their organ system insufficiency than elective surgical admissions. Because of this, 5 points are assigned to a patient with an existing chronic condition that is a non-operative or emergency postoperative patient, and 2 points are assigned for elective postoperative patients.

APACHE II MACRO

In order to use the macro to calculate the APACHE II score, the data must be formatted properly, and the required fields cannot contain missing values. This section will highlight the variables necessary to use the macro, how to implement the macro, and how to interpret the final APACHE II score in terms of predicted mortality.

GETTING THE DATA READY

The dataset to be used in the macro needs to have the following variables: patient age, if the patient had one of the five prior chronic health conditions, whether the patient was non-operative, emergency postoperative, or elective postoperative, if the patient had acute renal failure, and the minimum and maximum values (with a few exceptions) of the 12 physiologic variables shown in Table 4Table 1.

One of the largest components of the score is the APS. All of these values must be measured. This eliminates the problem of missing values and the concern about whether or not the missing value was normal. Special care should be used in only using measurements within the 24-hour time frame for the APS and in ensuring the correct unit for the measurement, such as temperature recorded in Celsius and not Fahrenheit. Minimum and maximum values are needed for the patient's temperature, mean arterial pressure, heart rate, respiratory rate, arterial pH, serum sodium, serum potassium, serum creatinine, hematocrit and white blood cell count. If the patient has no arterial pH measurements, serum HCO₃ can be used instead, although not preferred. For serum creatinine, if the patient has acute renal failure, the points associated with the value are doubled because loss of renal function is seen as a poor prognostic sign.

For oxygenation and the GCS, the measurements needed are different. GCS measures consciousness in an individual, its maximum value is 15, and a lower value signifies less conscious function. Because of this, only the lowest GCS within 24 hours is used in the calculation of the APACHE II score. For oxygenation, PO₂ levels are heavily dependent on inspired (FiO₂) O₂ levels, so there are different weightings for PO₂ levels based on the level of FiO₂. Only the maximum FiO₂ value and minimum PO₂ value are needed, but if FiO₂ is ≥ 0.50, the PCO₂ value at the same time PO₂ is the lowest is needed to calculate the patient's Aa-DO₂ in order to assign points for oxygenation levels.

For a full table of variable inputs with descriptions and units for the macro, see in the Table 5 in the Appendix.

CALCULATING THE APACHE II RAW SCORE

Once the dataset contains all values needed to calculate the score, the macro can be invoked to calculate the total APACHE II score. All the variables being input into the macro should be numeric. The macro contains one data step that uses series of if-then-else statements to assign points based on the variable value, to determine which measurements are more extreme, and to subsequently add the points together to give one summary score.

Using a sample dataset of 477 real admissions to the ICU floors at our organization, we will highlight different portions of the macro in the calculation of the APACHE II score. For the full macro, see the Appendix.

This code is used to calculate the points associated with patient temperature:

```
*temperature;
if (&tempmin >= 41) or (&tempmin <= 29.9) then tempmin_pts = 4;
else if (&tempmin >= 39 and &tempmin <= 40.9) or
      (&tempmin >= 30 and &tempmin <= 31.9) then tempmin_pts = 3;
else if (&tempmin >= 32 and &tempmin <= 33.9) then tempmin_pts = 2;
else if (&tempmin >= 38.5 and &tempmin <= 38.9) or
      (&tempmin >= 34 and &tempmin <= 35.9) then tempmin_pts = 1;
else if (&tempmin >= 36 and &tempmin <= 38.4) then tempmin_pts = 0;

if (&tempmax >= 41) or (&tempmax <= 29.9) then tempmax_pts = 4;
else if (&tempmax >= 39 and &tempmax <= 40.9) or
      (&tempmax >= 30 and &tempmax <= 31.9) then tempmax_pts = 3;
else if (&tempmax >= 32 and &tempmax <= 33.9) then tempmax_pts = 2;
else if (&tempmax >= 38.5 and &tempmax <= 38.9) or
      (&tempmax >= 34 and &tempmax <= 35.9) then tempmax_pts = 1;
else if (&tempmax >= 36 and &tempmax <= 38.4) then tempmax_pts = 0;

if tempmin_pts >= tempmax_pts then apache_score = apache_score + tempmin_pts;
else if tempmin_pts < tempmax_pts then apache_score = apache_score + tempmax_pts;
```

The minimum and maximum measurements are both assigned point values and then compared to find out which value is more extreme. The more extreme points are then added into the APACHE II score. Figure 1 shows the points assigned for the first five patients. Patient number 2 would have 3 points added into their APACHE II score for temperature, because the maximum temperature was seen as their most extreme value within 24 hours of ICU admission.

Obs	temp_min	temp_max	tempmin_pts	tempmax_pts
1	32.5	36.6	2	0
2	35.4	40.0	1	3
3	36.0	37.6	0	0
4	38.1	39.6	0	3
5	36.5	37.6	0	0

Figure 1 Temperature Points

This same format of code is used for the other APS variables that require minimum and maximum values (mean arterial pressure, heart rate, respiratory rate, serum sodium, serum potassium, serum creatinine, hematocrit and white blood cell count). The section of code for serum creatinine also includes statements to double the points if the patient has acute renal failure.

For arterial pH, sometimes patients will not have blood gas tests, so bicarbonate (HCO₃) measurements can be used instead. The following code shows how the macro will use the HCO₃ measurements if both the minimum and maximum pH values are missing:

```
*arterial pH/serum HCO3;
if (&artphmin = . and &artphmax = .) then do;
  if &hco3min >= 52 or &hco3min < 15 then hcomin_pts = 4;
  else if (&hco3min >= 41 and &hco3min <= 51.9) or
    (&hco3min >= 15 and &hco3min <= 17.9) then hcomin_pts = 3;
  else if (&hco3min >= 18 and &hco3min <= 21.9) then hcomin_pts = 2;
  else if (&hco3min >= 32 and &hco3min <= 40.9) then hcomin_pts = 1;
  else if (&hco3min >= 22 and &hco3min <= 31.9) then hcomin_pts = 0;

  if (&hco3max >= 52) or (&hco3max < 15) then hcomax_pts = 4;
  else if (&hco3max >= 41 and &hco3max <= 51.9) or
    (&hco3max >= 15 and &hco3max <= 17.9) then hcomax_pts = 3;
  else if (&hco3max >= 18 and &hco3max <= 21.9) then hcomax_pts = 2;
  else if (&hco3max >= 32 and &hco3max <= 40.9) then hcomax_pts = 1;
  else if (&hco3max >= 22 and &hco3max <= 31.9) then hcomax_pts = 0;

  if hcomin_pts >= hcomax_pts then apache_score = apache_score + hcomin_pts;
  else if hcomin_pts < hcomax_pts then apache_score = apache_score + hcomax_pts;
end;

else do;
  if (&artphmin >= 7.7) or (&artphmin < 7.15) then artphmin_pts = 4;
  else if (&artphmin >= 7.6 and &artphmin <= 7.69) or
    (&artphmin >= 7.15 and &artphmin <= 7.24) then artphmin_pts = 3;
  else if (&artphmin >= 7.25 and &artphmin <= 7.32) then artphmin_pts = 2;
  else if (&artphmin >= 7.5 and &artphmin <= 7.59) then artphmin_pts = 1;
  else if (&artphmin >= 7.33 and &artphmin <= 7.49) then artphmin_pts = 0;

  if (&artphmax >= 7.7) or (&artphmax < 7.15) then artphmax_pts = 4;
  else if (&artphmax >= 7.6 and &artphmax <= 7.69) or
    (&artphmax >= 7.15 and &artphmax <= 7.24) then artphmax_pts = 3;
  else if (&artphmax >= 7.25 and &artphmax <= 7.32) then artphmax_pts = 2;
  else if (&artphmax >= 7.5 and &artphmax <= 7.59) then artphmax_pts = 1;
  else if (&artphmax >= 7.33 and &artphmax <= 7.49) then artphmax_pts = 0;

  if artphmin_pts >= artphmax_pts then apache_score = apache_score + artphmin_pts;
  else if artphmin_pts < artphmax_pts then
    apache_score = apache_score + artphmax_pts;
end;
```

The first line checks to see if both the minimum and maximum arterial pH values are missing. If this is true, it will execute code that assigns points for both the minimum and maximum HCO₃ values. If at least one arterial pH value is present, the macro will assign points based on arterial pH.

Points associated with oxygenation require different logic. The macro will first check if the level of FiO₂ is above or equal to 0.50. If this is true, the macro will calculate the patient's A-aDO₂ level and assign points based on the A-aDO₂ level. Otherwise, the points associated with oxygenation are calculated with PO₂ values. The following code shows how points are assigned based on the level of FiO₂:

```
*oxygenation;
if &fio2max >= 0.5 then do;
  aado2 = (699.3*(&fio2max)) - (&po2min / 0.8) - (&pco2);
  if aado2 >= 500 then oxygen_pts = 4;
  else if aado2 >= 350 and aado2 <= 499 then oxygen_pts = 3;
  else if aado2 >= 200 and aado2 <= 349 then oxygen_pts = 2;
end;
```

```

else if aado2 < 200 then oxygen_pts = 0;
end;

else do;
  if &po2min > 70 then oxygen_pts = 0;
  else if &po2min >= 61 and &po2min <= 70 then oxygen_pts = 1;
  else if &po2min >= 55 and &po2min <= 60 then oxygen_pts = 3;
  else if &po2min < 55 then oxygen_pts = 4;
end;

apache_score = apache_score + oxygen_pts;

```

Figure 2 shows points associated with example patients who were below and above the FiO₂ cutoff of 0.50. The patients with FiO₂ levels above 0.50 have their A-aDO₂ calculated and have points assigned using that value, whereas patients with FiO₂ levels under 0.50 have points assigned based on PO₂ values.

<i>Obs</i>	<i>oxyfio2_max</i>	<i>po2_min</i>	<i>pco2</i>	<i>aado2</i>	<i>oxygen_pts</i>
2	0.8	113	26	392.19	3
11	0.8	159	24	336.69	2
20	0.4	96	38	.	0
21	0.4	71	31	.	0
22	0.4	116	49	.	0

Figure 2 Oxygenation Points

The calculation of the points associated with GCS, age, and chronic health conditions is fairly straightforward and is done with the following code:

```

*glasgow coma score;
gcs_pts = 15 - &gcsmin;

apache_score = apache_score + gcs_pts;

*age points;
if &age < 45 then age_pts = 0;
else if &age >= 45 and &age < 55 then age_pts = 2;
else if &age >= 55 and &age < 65 then age_pts = 3;
else if &age >= 65 and &age < 75 then age_pts = 5;
else if &age >= 75 then age_pts = 6;

apache_score = apache_score + age_pts;

*chronic health conditions;
if &chc = 0 then chc_pts = 0;
else if &chc = 1 and &surg in(0 1) then chc_pts = 5;
else if &chc = 1 and &surg = 2 then chc_pts = 2;

apache_score = apache_score + chc_pts;

```

Points for GCS are calculated by subtracting the minimum value from 15. Age points are assigned using if-then-else statements, and points for chronic health conditions are assigned based on the presence of a prior chronic health condition and the type of surgical admission. Figure 3 shows the point assignments for the first five patients. Patient number 1 would have a total of 23 points added to their APACHE II score: 12 for GCS, 6 for age, and 5 for having a prior health condition and being an emergent surgical admission.

<i>Obs</i>	<i>gcs_min</i>	<i>gcs_pts</i>	<i>age</i>	<i>age_pts</i>	<i>chp</i>	<i>surg</i>	<i>chc_pts</i>
1	3	12	83	6	1	1	5
2	3	12	66	5	0	.	0
3	3	12	84	6	0	.	0
4	7	8	62	3	0	.	0
5	9	6	81	6	1	1	5

Figure 3 GCS, Age, and Chronic Health Condition Points

After using the entire macro, variables for the possible points for each component of the tool and the total APACHE II summed score will be created in the dataset. Using the sample of ICU patients, Figure 4 shows the distribution of the final APACHE II scores calculated with our macro.

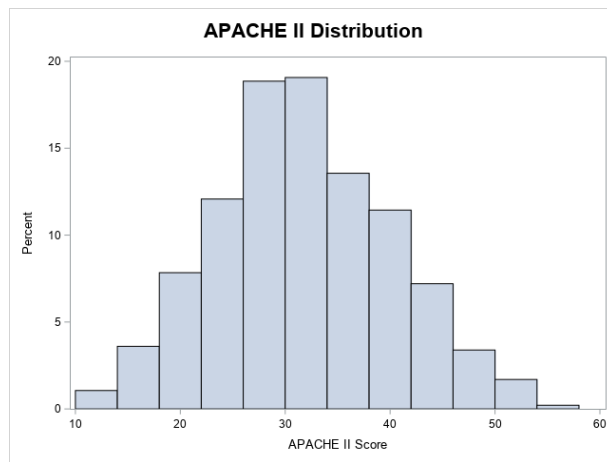


Figure 4 APACHE II Scores

INTERPRETATION OF THE SCORE

The APACHE II score can range from 0-71, where an increase in score is correlated with an increase in death rate. Table 3 Death Rate shows prognostic death rates for patients associated with ranges of the APACHE II score.

Score	Death Rate %
0-4	4
5-9	8
10-14	15
15-19	25
20-24	40
25-29	55
30-34	75
> 34	85

Table 3 Death Rates

CONCLUSION

Investigators in ICU research use the APACHE II score to compare the similarity of treatment and control groups. We set out to automate the calculation of the APACHE II

score and with this macro we are able to remove the manual burden of calculating the score for research purposes, but still provide a meaningful way to compare the two groups.

REFERENCES

Aakre, C.A., Dziadzko, M.A. & Herasevich, V. (2017, March). Towards automated calculation of evidence-based clinical scores. *World Journal of Methodology*, 7(1), 16-24.

Knaus, W.A., Draper, E.A., Wagner, D.P. & Zimmerman, J.E. (1985). APACHE II: a severity of disease classification system. *Critical Care Medicine*, 13, 818-829.

Office of the National Coordinator for Health Information Technology. (2017). Non-federal acute care hospital electronic health record adoption, Health IT Quick-Stat #47. dashboard.healthit.gov/quickstats/pages/FIG-Hospital-EHR-Adoption.php.

Rapsang, A. G., & Shyam, D. C. (2014, April). Scoring systems in the intensive care unit: A compendium. *Indian Journal of Critical Care Medicine*, 18(4), 220-228.

ACKNOWLEDGMENTS

The entire Scientific Support Team at Spectrum Health Office of Research: Nick Andersen, Nick Duesbery, Paul Egeler, Danielle Gritters, Cuyler Huffman, Andy Kampfschulte, Matt Lypka, Daniel Muzyka, and Derek Nedveck.

RECOMMENDED READING

Knaus, W.A., Draper, E.A., Wagner, D.P. & Zimmerman, J.E. (1985). APACHE II: a severity of disease classification system. *Critical Care Medicine*, 13, 818-829.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Margaret M. Kline
Grand Valley State University
Spectrum Health Office of Research
klinem@mail.gvsu.edu

Jessica L. Parker
Spectrum Health Office of Research
Jessica.Parker2@spectrumhealth.org

APPENDIX

Physiologic Variable	High Abnormal Range					Low Abnormal Range			
	+4	+3	+2	+1	0	+1	+2	+3	+4
Temperature – rectal (°C)	≥41	39 to 40.9		38.5 to 38.9	36 to 38.4	34 to 35.9	32 to 33.9	30 to 31.9	≤29.9
Mean Arterial Pressure (mm Hg)	≥160	130 to 159	110 to 129		70 to 109		50 to 69		≤49
Heart Rate (ventricular response)	≥180	140 to 179	110 to 139		70 to 109		55 to 69	40 to 54	≤39
Respiratory Rate (non-ventilated or ventilated)	≥50	35 to 49		25 to 34	12 to 24	10 to 11	6 to 9		≤5
Oxygenation: A-aDO ₂ or PaO ₂ (mm Hg) a. FIO ₂ ≥0.5 record A-aDO ₂ b. FIO ₂ <0.5 record PaO ₂	≥500	350 to 499	200 to 349		<200				
					PO ₂ >70	PO ₂ 61 to 70		PO ₂ 55 to 60	PO ₂ <55
Arterial pH (preferred)	≥7.7	7.6 to 7.69		7.5 to 7.59	7.33 to 7.49		7.25 to 7.32	7.15 to 7.24	<7.15
Serum HCO ₃ (venous mEq/l) (not preferred but may use if no ABGs)	≥52	41 to 51.9		32 to 40.9	22 to 31.9		18 to 21.9	15 to 17.9	<15
Serum Sodium (mEq/l)	≥180	160 to 179	155 to 159	150 to 154	130 to 149		120 to 129	111 to 119	≤110
Serum Potassium (mEq/l)	≥7	6 to 6.9		5.5 to 5.9	3.5 to 5.4	3 to 3.4	2.5 to 2.9		<2.5
Serum Creatinine (mg/dl) Double point score for acute renal failure	≥3.5	2 to 3.4	1.5 to 1.9		0.6 to 1.4		<0.6		
Hematocrit (%)	≥60		50 to 59.9	46 to 49.9	30 to 45.9		20 to 29.9		<20
White Blood Count (total/mm ³) (in 1000s)	≥40		20 to 39.9	15 to 19.9	3 to 14.9		1 to 2.9		<1
Glasgow Coma Score (GCS) Score = 15 minus actual GCS									

Table 4 APS Weighting System

Parameter	Description	Unit
DATASET	The dataset that includes all the variables	
TEMPMIN	The variable that contains the lowest temperature recorded in the first 24 hours of ICU admission	°C
TEMPMAX	The variable that contains the highest temperature recorded in the first 24 hours of ICU admission	°C
ARTPREMIN	The variable that contains the lowest mean arterial pressure recorded in the first 24 hours of ICU admission	mmHg
ARTPREMAX	The variable that contains the highest mean arterial pressure recorded in the first 24 hours of ICU admission	mmHg
HRTRATEMIN	The variable that contains the lowest heart rate recorded in the first 24 hours of ICU admission	Ventricular response
HRTRATEMAX	The variable that contains the highest heart rate recorded in the first 24 hours of ICU admission	Ventricular response
RESPRATEMIN	The variable that contains the lowest respiratory rate recorded in the first 24 hours of ICU admission	Non-ventilated or ventilated
RESPRATEMAX	The variable that contains the highest respiratory rate recorded in the first 24 hours of ICU admission	Non-ventilated or ventilated
FIO2MAX	The variable that contains the highest FiO ₂ value recorded in the first 24 hours of ICU admission	
PO2MIN	The variable that contains the lowest arterial PaO ₂ value recorded in the first 24 hours of ICU admission	
PCO2	The variable that contains the value of PaCO ₂ at the same time PaO ₂ is the lowest	
ARTPHMIN	The variable that contains the lowest arterial pH recorded in the first 24 hours of ICU admission	
ARTPHMAX	The variable that contains the highest arterial pH recorded in the first 24 hours of ICU admission	
HCO3MIN	Only needed if arterial pH is missing; the variable that contains the lowest bicarbonate (HCO ₃) recorded in the first 24 hours of ICU admission	
HCO3MAX	Only needed if arterial pH is missing; the variable that contains the highest bicarbonate (HCO ₃) recorded in the first 24 hours of ICU admission	
SODMIN	The variable that contains the lowest serum sodium recorded in the first 24 hours of ICU admission	mEq/l
SODMAX	The variable that contains the highest serum sodium recorded in the first 24 hours of ICU admission	mEq/l
POTASMIN	The variable that contains the lowest serum potassium recorded in the first 24 hours of ICU	mEq/l

	admission	
POTASMAX	The variable that contains the highest serum potassium recorded in the first 24 hours of ICU admission	mEq/l
CREATMIN	The variable that contains the lowest serum creatinine recorded in the first 24 hours of ICU admission	mg/dl
CREATMAX	The variable that contains the highest serum creatinine recorded in the first 24 hours of ICU admission	mg/dl
HCTMIN	The variable that contains the lowest hematocrit recorded in the first 24 hours of ICU admission	%
HCTMAX	The variable that contains the highest hematocrit recorded in the first 24 hours of ICU admission	%
WBCMIN	The variable that contains the lowest white blood cell count recorded in the first 24 hours of ICU admission	total/mm ³ (in 1000s)
WBCMAX	The variable that contains the highest white blood cell count recorded in the first 24 hours of ICU admission	total/mm ³ (in 1000s)
GCSMIN	The variable that contains the lowest Glasgow Coma Score recorded in the first 24 hours of ICU admission	
AGE	Patient's age	years
RENALFAIL	Presence or absence of acute renal failure; 0 = no, 1 = yes	
CHC	Presence or absence of prior chronic health condition; 0 = none, 1 = liver, cardiac, respiratory, or renal insufficiency or immunocompromised	
SURG	Type of admission; 0 = non-operative, 1 = emergent surgery, 2 = elective surgery	

Table 5 Macro Inputs

```

/*****
filename:  APACHE II Calculation.sas

purpose:   calculation of the APACHE II disease severity score

author:    margaret m. kline b.s., jessica l. parker, m.s.

date:      16jan2019

format:    %apacheII(dataset=, tempmin=, tempmax=, artpremin=, artpremax=,
           hrtratemin=, hrtratemax=, respratemin=, respratemax=,
           fio2max=, po2min=, pco2=, artphmin=, artphmax=,
           hco3min=, hco3max=, sodmin=, sodmax =, potasmin=, potasmax=,
           creatmin=, creatmax=, hctmin=, hctmax=, wbcmin=, wbcmax=,
           gcsmin=, age=, chc=, surg=, renalfail=,);

   required
   parameters:  dataset- dataset containing all necessary variables
                tempmin, tempmax- lowest and highest temperature within 24 hours of
ICU admission
                artpremin, artpremax- lowest and highest arterial blood pressure
within 24 hours of ICU admission
                hrtratemin, hrtratemax- lowest and highest heart rate within 24
hours of ICU admission
                respratemin, respratemax- lowest and highest respiratory rate
within 24 hours of ICU admission
                fio2max, po2min, pco2- highest FiO2 and lowest PO2 within 24 hours
of ICU admission,
                PCO2 value at the same
time PO2 is the lowest
                artphmin, artphmax- lowest and highest arterial pH within 24 hours
of ICU admission
                hco3min, hco3max- lowest and highest HCO3 within 24 hours of ICU
admission,
                only necessary if missing
                sodmin, sodmax- lowest and highest serum sodium within 24 hours of
ICU admission
                potasmin, potasmax- lowest and highest serum potassium within 24
hours of ICU admission
                creatmin, creatmax- lowest and highest serum creatinine within 24
hours of ICU admission
                hctmin, hctmax- lowest and highest hematocrit within 24 hours of
ICU admission
                wbcmin, wbcmax- lowest and highest WBC within 24 hours of ICU
admission
                gcsmin- lowest Glasgow Coma Score within 24 hours of ICU admission
age- patient age in years
                chc- presence of a prior chronic health condition;
                0=none, 1 = liver, cardiac, respiratory, renal or
immunocompromised
                surg- type of admission; 0 = non-operative, 1 = emergent surgery, 2
= elective surgery
                renalfail- presence of renal failure; 0 = no, 1 = yes
*****/

%macro apacheII(dataset=, tempmin=, tempmax=, artpremin=, artpremax=,
                hrtratemin=, hrtratemax=, respratemin=, respratemax=,
                fio2max=, po2min=, pco2=, artphmin=, artphmax=,
                hco3min=, hco3max=, sodmin=, sodmax =, potasmin=, potasmax=,
                creatmin=, creatmax=, hctmin=, hctmax=, wbcmin=, wbcmax=,
                gcsmin=, age=, chc=, surg=, renalfail=,);

data &dataset;
    set &dataset;
    apache_score = 0;

    *acute physiology score;
    *temperature;
    if (&tempmin >= 41) or (&tempmin <= 29.9) then tempmin_pts = 4;

```



```

    else if (&tempmin >= 39 and &tempmin <= 40.9) or (&tempmin >= 30 and &tempmin <= 31.9)
then tempmin_pts = 3;
    else if (&tempmin >= 32 and &tempmin <= 33.9) then tempmin_pts = 2;
    else if (&tempmin >= 38.5 and &tempmin <= 38.9) or (&tempmin >= 34 and &tempmin <= 35.9)
then tempmin_pts = 1;
    else if (&tempmin >= 36 and &tempmin <= 38.4) then tempmin_pts = 0;

    if (&tempmax >= 41) or (&tempmax <= 29.9) then tempmax_pts = 4;
    else if (&tempmax >= 39 and &tempmax <= 40.9) or (&tempmax >= 30 and &tempmax <= 31.9)
then tempmax_pts = 3;
    else if (&tempmax >= 32 and &tempmax <= 33.9) then tempmax_pts = 2;
    else if (&tempmax >= 38.5 and &tempmax <= 38.9) or (&tempmax >= 34 and &tempmax <= 35.9)
then tempmax_pts = 1;
    else if (&tempmax >= 36 and &tempmax <= 38.4) then tempmax_pts = 0;

    if tempmin_pts >= tempmax_pts then apache_score = apache_score + tempmin_pts;
    else if tempmin_pts < tempmax_pts then apache_score = apache_score + tempmax_pts;

*mean arterial pressure;
    if (&artpremin >= 160) or (&artpremin <= 49) then artpremin_pts = 4;
    else if (&artpremin >= 130 and &artpremin <= 159) then artpremin_pts = 3;
    else if (&artpremin >= 110 and &artpremin <= 129) or (&artpremin >= 50 and &artpremin <=
69) then artpremin_pts = 2;
    else if (&artpremin >= 70 and &artpremin <= 109) then artpremin_pts = 0;

    if (&artpremax >= 160) or (&artpremax <= 49) then artpremax_pts = 4;
    else if (&artpremax >= 130 and &artpremax <= 159) then artpremax_pts = 3;
    else if (&artpremax >= 110 and &artpremax <= 129) or (&artpremax >= 50 and &artpremax <=
69) then artpremax_pts = 2;
    else if (&artpremax >= 70 and &artpremax <= 109) then artpremax_pts = 0;

    if artpremin_pts >= artpremax_pts then apache_score = apache_score + artpremin_pts;
    else if artpremin_pts < artpremax_pts then apache_score = apache_score + artpremax_pts;

*heart rate;
    if (&hrtratemin >= 180) or (&hrtratemin <= 39) then hrtratemin_pts = 4;
    else if (&hrtratemin >= 140 and &hrtratemin <= 179) or (&hrtratemin >= 40 and &hrtratemin
<= 54) then hrtratemin_pts = 3;
    else if (&hrtratemin >= 110 and &hrtratemin <= 139) or (&hrtratemin >= 55 and &hrtratemin
<= 69) then hrtratemin_pts = 2;
    else if (&hrtratemin >= 70 and &hrtratemin <= 109) then hrtratemin_pts = 0;

    if (&hrtratemax >= 180) or (&hrtratemax <= 39) then hrtratemax_pts = 4;
    else if (&hrtratemax >= 140 and &hrtratemax <= 179) or (&hrtratemax >= 40 and &hrtratemax
<= 54) then hrtratemax_pts = 3;
    else if (&hrtratemax >= 110 and &hrtratemax <= 139) or (&hrtratemax >= 55 and &hrtratemax
<= 69) then hrtratemax_pts = 2;
    else if (&hrtratemax >= 70 and &hrtratemax <= 109) then hrtratemax_pts = 0;

    if hrtratemin_pts >= hrtratemax_pts then apache_score = apache_score + hrtratemin_pts;
    else if hrtratemin_pts < hrtratemax_pts then apache_score = apache_score +
hrtratemax_pts;

*respiratory rate;
    if (&respratemin >= 50) or (&respratemin <= 5) then respratemin_pts = 4;
    else if (&respratemin >= 35 and &respratemin <= 49) then respratemin_pts = 3;
    else if (&respratemin >= 6 and &respratemin <= 9) then respratemin_pts = 2;
    else if (&respratemin >= 25 and &respratemin <= 34) or (&respratemin >= 10 and
&respratemin <= 11) then respratemin_pts = 1;
    else if (&respratemin >= 12 and &respratemin <= 24) then respratemin_pts = 0;

    if (&respratemax >= 50) or (&respratemax <= 5) then respratemax_pts = 4;
    else if (&respratemax >= 35 and &respratemax <= 49) then respratemax_pts = 3;
    else if (&respratemax >= 6 and &respratemax <= 9) then respratemax_pts = 2;
    else if (&respratemax >= 25 and &respratemax <= 34) or (&respratemax >= 10 and
&respratemax <= 11) then respratemax_pts = 1;
    else if (&respratemax >= 12 and &respratemax <= 24) then respratemax_pts = 0;

    if respratemin_pts >= respratemax_pts then apache_score = apache_score + respratemin_pts;
    else if respratemin_pts < respratemax_pts then apache_score = apache_score +
respratemax_pts;

```

```

*oxygenation;
if &fio2max >= 0.5 then do;
    aado2 = (699.3*(&fio2max)) - (&po2min / 0.8) - (&pco2);
    if aado2 >= 500 then oxygen_pts = 4;
    else if aado2 >= 350 and aado2 <= 499 then oxygen_pts = 3;
    else if aado2 >= 200 and aado2 <= 349 then oxygen_pts = 2;
    else if aado2 < 200 then oxygen_pts = 0;
end;

else do;
    if &po2min > 70 then oxygen_pts = 0;
    else if &po2min >= 61 and &po2min <= 70 then oxygen_pts = 1;
    else if &po2min >= 55 and &po2min <= 60 then oxygen_pts = 3;
    else if &po2min < 55 then oxygen_pts = 4;
end;

apache_score = apache_score + oxygen_pts;

*arterial pH/serum HCO3;
if (&artphmin = . and &artphmax = .) then do;
    if &hco3min >= 52 or &hco3min < 15 then hcomin_pts = 4;
    else if (&hco3min >= 41 and &hco3min <= 51.9) or (&hco3min >= 15 and &hco3min <=
17.9) then hcomin_pts = 3;
    else if (&hco3min >= 18 and &hco3min <= 21.9) then hcomin_pts = 2;
    else if (&hco3min >= 32 and &hco3min <= 40.9) then hcomin_pts = 1;
    else if (&hco3min >= 22 and &hco3min <= 31.9) then hcomin_pts = 0;

    if (&hco3max >= 52) or (&hco3max < 15) then hcomax_pts = 4;
    else if (&hco3max >= 41 and &hco3max <= 51.9) or (&hco3max >= 15 and &hco3max <=
17.9) then hcomax_pts = 3;
    else if (&hco3max >= 18 and &hco3max <= 21.9) then hcomax_pts = 2;
    else if (&hco3max >= 32 and &hco3max <= 40.9) then hcomax_pts = 1;
    else if (&hco3max >= 22 and &hco3max <= 31.9) then hcomax_pts = 0;

    if hcomin_pts >= hcomax_pts then apache_score = apache_score + hcomin_pts;
    else if hcomin_pts < hcomax_pts then apache_score = apache_score + hcomax_pts;
end;

else do;
    if (&artphmin >= 7.7) or (&artphmin < 7.15) then artphmin_pts = 4;
    else if (&artphmin >= 7.6 and &artphmin <= 7.69) or (&artphmin >= 7.15 and
&artphmin <= 7.24) then artphmin_pts = 3;
    else if (&artphmin >= 7.25 and &artphmin <= 7.32) then artphmin_pts = 2;
    else if (&artphmin >= 7.5 and &artphmin <= 7.59) then artphmin_pts = 1;
    else if (&artphmin >= 7.33 and &artphmin <= 7.49) then artphmin_pts = 0;

    if (&artphmax >= 7.7) or (&artphmax < 7.15) then artphmax_pts = 4;
    else if (&artphmax >= 7.6 and &artphmax <= 7.69) or (&artphmax >= 7.15 and
&artphmax <= 7.24) then artphmax_pts = 3;
    else if (&artphmax >= 7.25 and &artphmax <= 7.32) then artphmax_pts = 2;
    else if (&artphmax >= 7.5 and &artphmax <= 7.59) then artphmax_pts = 1;
    else if (&artphmax >= 7.33 and &artphmax <= 7.49) then artphmax_pts = 0;

    if artphmin_pts >= artphmax_pts then apache_score = apache_score + artphmin_pts;
    else if artphmin_pts < artphmax_pts then apache_score = apache_score +
artphmax_pts;
end;

*serum sodium;
if (&sodmin >= 180) or (&sodmin <= 110) then sodmin_pts = 4;
else if (&sodmin >= 160 and &sodmin <= 179) or (&sodmin >= 111 and &sodmin <= 119) then
sodmin_pts = 3;
else if (&sodmin >= 155 and &sodmin <= 159) then sodmin_pts = 2;
else if (&sodmin >= 150 and &sodmin <= 154) then sodmin_pts = 1;
else if (&sodmin >= 130 and &sodmin <= 149) then sodmin_pts = 0;

if (&sodmax >= 180) or (&sodmax <= 110) then sodmax_pts = 4;
else if (&sodmax >= 160 and &sodmax <= 179) or (&sodmax >= 111 and &sodmax <= 119) then
sodmax_pts = 3;
else if (&sodmax >= 155 and &sodmax <= 159) then sodmax_pts = 2;

```

```

else if (&sodmax >= 150 and &sodmax <= 154) then sodmax_pts = 1;
else if (&sodmax >= 130 and &sodmax <= 149) then sodmax_pts = 0;

if sodmin_pts >= sodmax_pts then apache_score = apache_score + sodmin_pts;
else if sodmin_pts < sodmax_pts then apache_score = apache_score + sodmax_pts;

*serum potassium;
if (&potasmin >= 7) or (&potasmin <= 2.5) then potasmin_pts = 4;
else if (&potasmin >= 6 and &potasmin <= 6.9) then potasmin_pts = 3;
else if (&potasmin >= 2.5 and &potasmin <= 2.9) then potasmin_pts = 2;
else if (&potasmin >= 5.5 and &potasmin <= 5.9) or (&potasmin >= 3 and &potasmin <= 3.4)
then potasmin_pts = 1;
else if (&potasmin >= 3.5 and &potasmin <= 5.4) then potasmin_pts = 0;

if (&potasmax >= 7) or (&potasmax <= 2.5) then potasmax_pts = 4;
else if (&potasmax >= 6 and &potasmax <= 6.9) then potasmax_pts = 3;
else if (&potasmax >= 2.5 and &potasmax <= 2.9) then potasmax_pts = 2;
else if (&potasmax >= 5.5 and &potasmax <= 5.9) or (&potasmax >= 3 and &potasmax <= 3.4)
then potasmax_pts = 1;
else if (&potasmax >= 3.5 and &potasmax <= 5.4) then potasmax_pts = 0;

if potasmin_pts >= potasmax_pts then apache_score = apache_score + potasmin_pts;
else if potasmin_pts < potasmax_pts then apache_score = apache_score + potasmax_pts;

*serum creatinine;
if (&creatmin >= 3.5) then creatmin_pts = 4;
else if (&creatmin >= 2 and &creatmin <= 3.4) then creatmin_pts = 3;
else if (&creatmin >= 1.5 and &creatmin <= 1.9) or (&creatmin < 0.6) then creatmin_pts =
2;
else if (&creatmin >= 0.6 and &creatmin <= 1.4) then creatmin_pts = 0;

if (&creatmax >= 3.5) then creatmax_pts = 4;
else if (&creatmax >= 2 and &creatmax <= 3.4) then creatmax_pts = 3;
else if (&creatmax >= 1.5 and &creatmax <= 1.9) or (&creatmax < 0.6) then creatmax_pts =
2;
else if (&creatmax >= 0.6 and &creatmax <= 1.4) then creatmax_pts = 0;

if &renalfail = 1 then do creatmin_pts = creatmin_pts * 2;
creatmax_pts = creatmax_pts * 2; end;

if creatmin_pts >= creatmax_pts then apache_score = apache_score + creatmin_pts;
else if creatmin_pts < creatmax_pts then apache_score = apache_score + creatmax_pts;

*hematocrit;
if (&hctmin >= 60) or (&hctmin < 20) then hctmin_pts = 4;
else if (&hctmin >= 50 and &hctmin <= 59.9) or (&hctmin >= 20 and &hctmin <= 29.9) then
hctmin_pts = 2;
else if (&hctmin >= 46 and &hctmin <= 49.9) then hctmin_pts = 1;
else if (&hctmin >= 30 and &hctmin <= 45.9) then hctmin_pts = 0;

if (&hctmax >= 60) or (&hctmax < 20) then hctmax_pts = 4;
else if (&hctmax >= 50 and &hctmax <= 59.9) or (&hctmax >= 20 and &hctmax <= 29.9) then
hctmax_pts = 2;
else if (&hctmax >= 46 and &hctmax <= 49.9) then hctmax_pts = 1;
else if (&hctmax >= 30 and &hctmax <= 45.9) then hctmax_pts = 0;

if hctmin_pts >= hctmax_pts then apache_score = apache_score + hctmin_pts;
else if hctmin_pts < hctmax_pts then apache_score = apache_score + hctmax_pts;

*white blood count;
if (&wbcmin >= 40) or (&wbcmin < 1) then wbcmin_pts = 4;
else if (&wbcmin >= 20 and &wbcmin <= 39.9) or (&wbcmin >= 1 and &wbcmin <= 2.9) then
wbcmin_pts = 2;
else if (&wbcmin >= 15 and &wbcmin <= 19.9) then wbcmin_pts = 1;
else if (&wbcmin >= 3 and &wbcmin <= 14.9) then wbcmin_pts = 0;

if (&wbcmax >= 40) or (&wbcmax < 1) then wbcmax_pts = 4;
else if (&wbcmax >= 20 and &wbcmax <= 39.9) or (&wbcmax >= 1 and &wbcmax <= 2.9) then
wbcmax_pts = 2;
else if (&wbcmax >= 15 and &wbcmax <= 19.9) then wbcmax_pts = 1;
else if (&wbcmax >= 3 and &wbcmax <= 14.9) then wbcmax_pts = 0;

```

```

if wbcmin_pts >= wbcmax_pts then apache_score = apache_score + wbcmin_pts;
else if wbcmin_pts < wbcmax_pts then apache_score = apache_score + wbcmax_pts;

*glasgow coma score;
gcs_pts = 15 - &gcsmin;
apache_score = apache_score + gcs_pts;

*age points;
if &age < 45 then age_pts = 0;
else if &age >= 45 and &age < 55 then age_pts = 2;
else if &age >= 55 and &age < 65 then age_pts = 3;
else if &age >= 65 and &age < 75 then age_pts = 5;
else if &age >= 75 then age_pts = 6;

apache_score = apache_score + age_pts;

*chronic health conditions;
if &chc = 0 then chc_pts = 0;
else if &chc = 1 and &surg in(0 1) then chc_pts = 5;
else if &chc = 1 and &surg = 2 then chc_pts = 2;

apache_score = apache_score + chc_pts;

run;

%mend apacheII;

```