Predicting Aircraft Maneuvers to Evaluate the Reliability of Non-Expert Human Observers

David R Crow, 2d Lt, USAF

***Abstract*—Often, flying organizations require accurate information about what their airplanes are doing. Applying labels to various maneuvers, however, is a nontrivial task. In this research, we utilize a flight simulator to generate data about an aircraft's orientation, altitude, airspeed, vertical velocity, and acceleration as the values change over time. Both an expert system and a non-expert human label each datapoint with a flight maneuver. We then fit machine learning models to the human's labels to determine whether the non-expert applies the same labels as the expert system. If the expert system's labels are significantly different from those of the best possible model, then we know that the layman-applied labels are not reliable. However, even if the non-expert's labels are unreliable, a demonstrated ability to fit a model to the labels may indicate an ability to replace expert labelers with models fit to the experts, thus saving hours of manpower.**

***Index Terms*—classification, aircraft flight dynamics, human reliability, supervised learning**

I. Introduction

An aircraft's maneuvers in flight are often directly indicated by the plane's roll, pitch, and yaw. Takeoff, for example, can be defined by a high, positive pitch and low, well-correlated values for roll and yaw. Humans, who are naturally error-prone, might misidentify an aircraft's movement when given only a visual observation of the plane. The United States Air Force (USAF) is often concerned with the movement of its various aircraft (and those of other countries), but, in situations without access to the aircraft's flight data (to include roll, pitch, and yaw), the Air Force must rely on human observers. Presently, the USAF does not know much about the reliability of non-expert human observers in identifying aircraft maneuvers.

In this research, multiple machine learning (ML) models are fit to a set of aircraft characteristics generated by a flight simulator. A non-expert human observer labels each of the simulated observations with the maneuver of the aircraft at that point in time; these labels serve as the model's truth data. We evaluate each model's performance and then identify which model best predicts the aircraft's maneuver. An expert system also labels the observations, and these labels are then compared to those predicted by the best model. In doing so, we can determine whether the human observer provides reliable testimony. Ideally, the results concerning reliability of observers generalize to other domains, even if those domains are not related to the military.

This research has a secondary objective: effective resource management. A demonstrated ability to accurately fit a model to *layman-defined* flight maneuvers implies an ability to accurately fit a model to *expert-defined* flight maneuvers. This is because the expert is likely to be much more consistent – and thus her labels are much less noisy – than the layman. This new model can be used to categorize flight maneuvers as an expert without wasting the time of an expert.

Many researchers have used flight characteristics to predict other flight characteristics, aircraft type, and even the maneuver at some point in time. However, previous research, as indicated in Section II, does not evaluate the reliability of its labels; instead, the researchers always assume their dataset is accurately labeled (when it is actually labeled; see [1]). In this research, ML models are fit to the dataset much like in previous research, but we go further in comparing the best model's performance to that of an expert system.

In general, it is expected that the best model will be able to successfully predict the non-expert's labels. The real significance of this research is in evaluating the correctness of the human-labeled maneuvers. It is certainly possible that the non-expert's definition of (say) turning is different from an expert's, and we wish to determine whether a model fit to the layman's definition is even useful in practical applications. Answering this question may illustrate why aircraft maneuvers labeled by non-experts are or are not reliable.

Regardless of whether the non-expert's labels are correct, accurately fitting a model to the dataset implies that a model can also be fit to an expert's labels. Such a model can reduce the use of human resources in categorizing flight data. Although the answer to this question is less impactful than the answer to the first question, it might still save the USAF considerable time and resources in the right situation.

The Air Force Research Lab (AFRL) maintains a flight simulator, the Avionics Vulnerability Assessment System (AVAS), which generates the dataset used in this research. At any given moment, the system computes various metrics, including airspeed, angle of attack, latitude, heading, and wind angle. This reseach concerns the airplane's orientation, speed, acceleration, and altitude. The simulator is able to display all values as they change over time, and recent changes to the source code enable parameter filtering and file output.

To most effectively fit a model to the dataset, we train various classifiers and conduct a performance analysis of each. The software program that generates the dataset enables fully-supervised learning; this allows us to easily determine the utility of each classifier. After fitting the best possible model to the non-expert-labeled dataset, its performance is compared to that of an expert system (which labels flight maneuvers based on the roll, pitch, and yaw of the aircraft) to evaluate the reliability of the non-expert labels and the overall utility of a model fit to a human's labels.

Results indicate that one can fit a model to a non-expert's labels with a high degree of correctness. Results also indicate

that such a model can not predict an expert's labels at the same performance level. One may conclude from these results that a layman is not a reliable source of maneuver labels. However, one may also conclude that a machine learning model trained on the expert's labels can likely replace the expert in labeling future observations.

In the remainder of this report, we present the research in detail. Section II examines some of the related work in current literature and explain why this work is insufficient for the research at hand. Section III describes the dataset, the model-fitting process, and the model analysis and evaluation tools. Section IV discusses the results obtained. Section V explains the implication of the results and possible opportunities for future research.

II. Related Work

The primary reference material for this research is [2]. Without it (and its accompanying resources), we likely would not have developed a sufficient understanding of relevant ML techniques. Specifically, [2] details dataset partitioning, the best subset and ridge feature selection and regularization methods, k-fold cross-validation (CV), binary classification, k-nearest neighbors (KNN) classification, classification foests, support vector classifiers (SVCs), and the evaluation of classifier performance.

When [2] proves insufficient, we reference [3]. This resource describes classification in greater detail, and it also explains several classification methods not covered in [2]. Additionally, [3] discusses multiple sampling methods not detailed elsewhere. Although we mostly utilize Python libraries for sampling, this reference provides insight into why other methods might be more appropriate.

The work performed in [4] is closely-aligned with our own research. The researchers utilize an open-source dataset consisting of worldwide flight data. After collecting flight characteristics from the dataset via robust feature engineering, they predict the aircraft's type. Clearly, this research is in the same domain as our own, and we thus refer to it when evaluating the models and reporting the performance. Note that the researchers are only concerned with objective labels (i.e., aircraft type), instead of the subjective labels predicted in this research.

Reference [1] is also related to our research. Like us, the researchers predict aircraft maneuvers, but they fit an artificial neural networks model to a partially-unknown dataset. However, the class labels – although more complex than our own – are already given by expert systems; there is little subjectivity or doubt in the labels (for those observations that actually have labels). This resource does not address the correctness of the labels themselves.

This research is necessary because previous research does not evaluate questionable class labels. In other words, it does not evaluate the reliability of humans in characterizing aircraft flight maneuvers. The following section details the dataset and its components: the generation process, the features, and the class balance. Additionally, it explores the various features and hypothesize about the best and worst predictors of airplane maneuvers.

III. Methodology

This section discusses the data generation process and the resulting features and observations. Additionally, it details the machine learning process. Specifically, it explains the model fitting procedure, to include partitioning, validation, and other ML design choices. The section also describes the evaluation procedure: how we compare different models, how we compute results, and how we analyze the results.

A. Data

The data are generated by the AVAS, an AFRL-developed military flight simulator that employs real-world physics and flight dynamics for USAF research purposes. To create the dataset, the simulator is repeatedly guided through takeoff and various midair maneuvers. The simulator generates about 500 observations per minute, so, by documenting the relative start and end times of a maneuver, the datapoints can be easily segregated and labeled. By turning for n seconds, taking off for n seconds, and flying (approximately) straight and level for n seconds, we can balance the distribution of the three classes and generate a sizable dataset. The AVAS source code was modified for this research so that it outputs the observations directly to a properly-formatted comma-separated values file; thus, the generated file is ready for ML.

Fortunately, the availability of the AVAS allows us to generate arbitrary amounts of data. For this research, $5,000$ observations constitute a sufficiently-large dataset. Each observation includes nine different flying metrics and a timestamp relative to the start of the simulation. The metrics are roll, pitch, and yaw (each in radians), altitude (in feet), airspeed and vertical velocity (in feet per second), and acceleration in each of three axes (in feet per second per second). The roll and pitch values range from $-\pi$ to π; yaw ranges from 0 to 2π; the altitude and airspeed are both greater than zero; the vertical velocity and accelerations are real number values. The timestamp data is only useful for labeling, so it is removed prior to fitting the models.

When flying the simulator, the maneuver performed during a given time period is noted, and all observations within this window are labeled with the maneuver in question; in doing so, the truth data is identified. Thus, the truth data are labeled according to the beliefs of the individual operating the simulator. Because this individual is not well-versed in aircraft flight, it is assumed that these labels are not always correct.

The dataset has three classes: *taking off*, *turning*, and *cruising*. By repeating the simulations with different starting conditions, we are able to balance the presence of each of the three classes in the dataset. Because the dataset is so large, we are also able to selectively filter the data to achieve different class distributions (if desired).

Fig. 1 shows nine different histograms, one for each feature. These histograms illustrate the distribution of the features over all observations. Roll, pitch, vertical velocity, y-axis acceleration, and z-axis acceleration are all relatively uniformly-distributed (perhaps with some skewing). The remaining features are not uniformly-distributed.

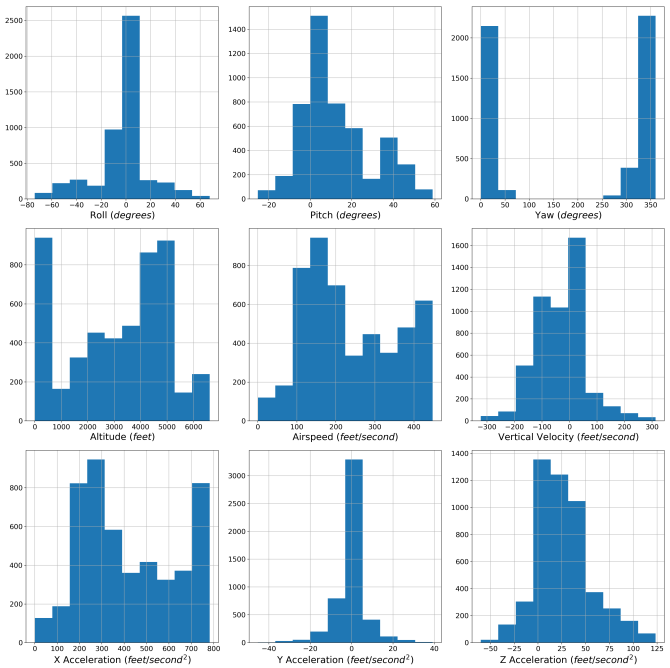

Fig. 1. A histogram for each feature. From left to right, top to bottom: roll, pitch, yaw, altitude, airspeed, vertical velocity, x-axis acceleration, y-axis acceleration, z-axis acceleration. The roll, pitch, and yaw values are shown in degrees to allow for greater interpretability.

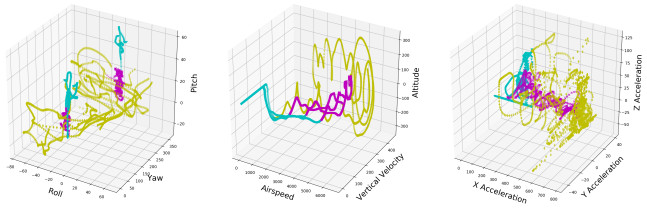

Fig. 2. Three-dimensional plots for groups of similar features. The takeoff, cruise, and turn classes are shown in cyan, magenta, and yellow, respectively. *Left*: roll vs. yaw vs. pitch. *Middle*: airspeed vs. vertical velocity vs. altitude. *Right*: x-axis acceleration vs. y-axis acceleration vs. z-axis acceleration.

Clearly, the values of the features cover significantly different ranges. For this reason, all features are scaled to ensure each feature is important in our model formulation. Scaling is completed in the data pre-processing phase.

In Fig. 2, each of three disparate groups of features are plotted on a three-dimensional (3D) plot. Each group contains related features; for example, the first group contains roll, pitch, and yaw. As we examine the plots from left to right, it is clear that the boundaries between the classes grow less distinct. Thus, we predict that the features in the left plot outperform those in the middle plot, and we expect those in the middle plot to outperform those in the right plot.

Additionally, it is expected that roll, pitch, and yaw are the best predictors of flight maneuvers specifically because the maneuvers in question are essentially *defined* by roll, pitch, and yaw. Fig. 2 supports this hypothesis: clearly, the three classes are most distinct in the left plot. As a secondary objective, then, we examine model performance when all information about the aircraft's orientation is excluded. It is not likely a model which does not use roll, pitch, or yaw data will perform as well as one that does, but we suspect that such a model can still adequately classify flight maneuvers.

B. Model Fitting

To compare the model's performance using layman-defined labels to the performance of one that uses expert-defined labels, we need to build the strongest possible classification model. For this reason, multiple classification approaches are evaluated: one-versus-all logistic regression [3], quadratic discriminant analysis (QDA), the ridge, the random forest classification approach, KNN classification, and support vector classification [2]. In all cases, of course, the fully-labeled dataset means that we conduct fully-supervised learning.

To effectively evaluate the performance of each model, we partition the dataset into training and testing sets using proportional random sampling. Specifically, the training set consists of 90% of the observations from each class; the remaining data points constitute the test set. Because the classes are balanced in the full dataset, they are also balanced in both the training and testing sets.

When building the various models, we use k-fold CV. This ensures that the models are not unnecessarily biased toward the training observations. For $5,000$ observations, $k = 10$ is appropriate. This approach certainly increases the model-building complexity (and hence the computation time), but this research requires the best model so that we can effectively evaluate the reliability of human observers; for this reason, we must minimize variance, and thus effective CV is necessary.

C. Feature Selection

It is certainly possible that some of the available features are not very relevant to maneuver prediction (however, we do not expect this to be the case). Thus, we also conduct feature selection and regularization using best subset selection and the ridge [2]. The best subset method is used to fit multiple logistic regression and QDA models. Various α values are used to fit multiple ridge models. By utilizing both approaches, the likelihood of identifying the best classification model is increased. The extra computational resources required to test two feature selection methods – instead of just one – are negligible. In other words, there is not much reason to *not* evaluate both techniques.

D. Iteration

For each type of classification model, we iterate over the model-fitting parameters until the optimal value is identified. Specifically, we iterate over every possible subset of the set of features for logistic regression and QDA; over α, the strength of the regularization, for ridge classification; over tree depth and m, the number of features to consider at each internal node, for the classification forest; over n, the number of neighbors to consider, for KNN; and over the type of kernel and C, the classifier's *budget*, for SVC. (If the SVC kernel is polynomial, we also iterate over the polynomial degree.)

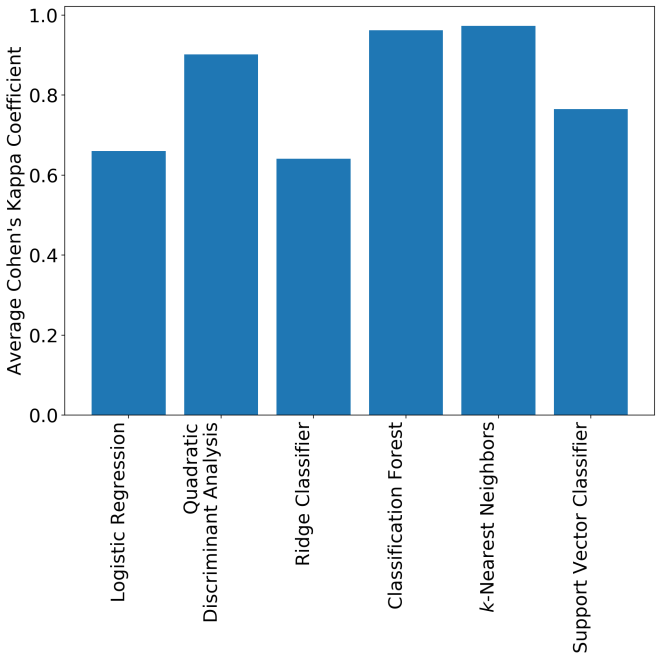

Fig. 3. Cohen's kappa coefficient averaged over all parameter settings for each of the six types of classifier. These are the κ values between models and the layman's labels, as computed on the training set during model fitting.

E. Model Evaluation & Analysis

By tuning the models and their parameters, the single best classifier for the dataset is identified. This optimal model is then compared to a simple, expert predictor based solely on roll, pitch, and yaw. Large discrepancies in performance of the two systems indicate that the layman's observations are mislabeled. In other words, large discrepancies imply that non-experts are not reliable in characterizing aircraft maneuvers. Small discrepancies, on the other hand, illustrate that the labels are accurate in most cases.

To actually identify the best model and compare it against the expert system, a reliable evaluation procedure is needed. In this research, we utilize Cohen's kappa coefficient [5] to easily compare model performance. For two sets of labels a and b, this value is defined by

$$\kappa = \frac{p_o - p_c}{1 - p_c} = 1 - \frac{1 - p_o}{1 - p_c},$$

where p_o is the observed agreement between a and b, and p_c is the chance agreement between a and b. More details can be found in [5]. We must compute this value for each of the models (that is, each of the model and parameter value combinations), but repeating these computations is trivial. The scores identify the best and worst models, and they also describe the level of agreement between the best model and the expert system.

We have now described how the classification performance of each model is evaluated. The following section details the results of the model evaluation. Specifically, it describes which models and parameter settings performed best, and it dicusses what each of various potential results might indicate about the system and about other, less-related systems.

TABLE I
COHEN'S KAPPA COEFFICIENT FOR EACH CLASSIFIER

Model	Layman	Expert
Logistic Regression	0.696015	0.677175
Quadratic Discriminant Analysis	0.957674	0.927529
Ridge Classification	0.657370	0.631847
Classification Forest	0.996973	0.900345
k-Nearest Neighbors	0.993946	0.897330
Support Vector Classification	0.993946	0.897330

TABLE II
ACCURACY FOR EACH CLASSIFIER

Model	Layman	Expert
Logistic Regression	0.802783	0.782977
Quadratic Discriminant Analysis	0.973443	0.951512
Ridge Classification	0.777559	0.752535
Classification Forest	0.997955	0.933480
k-Nearest Neighbors	0.996134	0.931484
Support Vector Classification	0.996134	0.931484

IV. RESULTS & DISCUSSION

This section describes the results achieved. Overall, the results indicate that the best models fit the non-expert's labels almost perfectly, but they also indicate – as expected – that the non-expert does not label maneuvers exactly like the expert.

Fig. 3 depicts the average performance of each of the six model types in fitting the layman's labels. Clearly, QDA, the classification forest, and KNN perform consistently well. As can be seen in Tables I, III, II, and IV, SVC can perform extremely well, but Fig. 3 shows that the poorest versions of SVC significantly decrease its average performance.

Table I contains the κ values between the best version of each model (e.g., KNN when $k = 1$ or the ridge when $\alpha = 1.274275$) and the non-expert's labels. The table also contains the κ value between the best version of each model and the expert's labels.

For completeness, and to compare Cohen's κ coefficient to statistical accuracy, the accuracy scores between the best version of each model and the non-expert's labels and between

TABLE III
COHEN'S KAPPA COEFFICIENT FOR EACH CLASSIFIER WHEN ROLL, PITCH, AND YAW FEATURES ARE EXCLUDED

Model	Layman	Expert
Logistic Regression	0.656764	0.637979
Quadratic Discriminant Analysis	0.739439	0.719577
Ridge Classification	0.587684	0.564931
Classification Forest	0.993946	0.897330
k-Nearest Neighbors	0.984867	0.894308
Support Vector Classification	0.993946	0.897330

TABLE IV
ACCURACY FOR EACH CLASSIFIER WHEN ROLL, PITCH, AND YAW FEATURES ARE EXCLUDED

Model	Layman	Expert
Logistic Regression	0.775670	0.756660
Quadratic Discriminant Analysis	0.835090	0.813799
Ridge Classification	0.729184	0.707838
Classification Forest	0.996134	0.931484
k-Nearest Neighbors	0.990283	0.929341
Support Vector Classification	0.996134	0.931484

the best version of each model and the expert's labels are shown in Table II. As expected, the accuracy scores are higher than the respective κ coefficients; this is because accuracy does not consider the possibility of chance agreement.

Tables III and IV are much like Tables I and II, but all models in Tables III and IV are fit without roll, pitch, and yaw data. Some restricted models, like the classification forest, KNN, and SVC, perform much like they do when orientation data is included. The remaining models – logistic regression, QDA, and the ridge – perform noticeably worse. Each restricted model is certainly able to fit the layman's labels better than the expert's, but, again, the difference in κ values is less than expected.

Depending on the type of model used, one can clearly fit a near-perfect model to the layman's labels, even if roll, pitch, and yaw data are excluded. When all features are included, the classification forest approach outperforms all others. We suspect this is due to the nature of the labels in question. One can easily rule out possible classes for a given observation using limited information. For example, a low roll value means that the aircraft is not turning; one can then easily evaluate altitude and airspeed to decide between *takeoff* and *cruise*.

A high-performing model, however, is not likely to predict the expert's labels quite as well. In fact, the best models are 10% more likely to correctly predict the layman's labels than they are to correctly predict the expert's labels. This may indicate a degree of unreliability in the layman's labels.

Still, because some classification models fit the layman's labels so well, and because an expert is likely to be more consistent than a layman in labeling maneuvers, we assert that a classification model can also fit an expert's labels. Such a model can then label flight data in place of expert labelers.

The final section discusses conclusions one can draw from the research and the results. Additionally, it considers potential avenues for future, related research.

V. Conclusions & Future Work

The results obtained in this research indicate two outcomes:

1) A human responsible for labeling aircraft data requires expert knowledge to accurately classify an aircraft's maneuvers; and
2) A machine learning model can adequately represent a human labeler of flight maneuvers, and thus such a model can classify large amounts of data and reduce manpower requirements.

The remainder of this section describes these conclusions in detail. Additionally, it discusses future research possibilities, including the use of a more robust simulator and a much larger number of flight maneuvers.

If an expert system's labels are significantly different[1] from those of the model fit to a layman's labels, we know that the layman's labels are unreliable. The results clearly show this to be the case for the best ML models[2]. In other words, the results detailed in Section IV indicate that the layman's label for a given observation is often *not* representative of the true aircraft maneuver for that observation. Thus, we can conclude that a human observer needs expert knowledge to accurately classify aircraft maneuvers. This conclusion likely applies to more than just aircraft maneuvers (for example, in evaluating the validity of crime-scene witness testimony), but further research is required to validate this claim.

Although the results show that a model fit to the layman's labels does not adequately predict the expert's labels, they also show that the model fits the layman's labels exceptionally well. The best model's kappa coefficient is 0.996973 (where 1 is the maximum possible value), so it is clear that the model is representative of the non-expert. It is assumed that an expert applies labels in a more consistent manner than does a non-expert, so we believe a machine learning model can fit an expert at least as well as it fits the layman. Thus, one could fit a model to an expert and then use such a model to classify large amounts of aircraft flight data (for whatever purpose is required). In doing so, one would reduce manpower requirements – at least for one's expert maneuver-labelers.

The AVAS, while useful for this research, is limited in scope. The control scheme is low-quality, the visual representation is sub-par, and the code is far too convoluted to allow for significant modifications. Future researchers who wish to affirm the conclusions found here should generate data with a higher-quality, professional simulator, or they should work to obtain real data from real aircraft. The latter is ideal.

Additionally, future research should analyze whether our conclusions hold when the number of classes is significantly large. A powered aircraft's movements can be classified by so much more than just *takeoff*, *cruise*, and *turn*, and future researchers would do well to apply more labels to their observations. Of course, the expert system would also need to be improved to allow for a greater number of classes.

This research examined the reliability of non-expert labelers of aircraft maneuvers. According to the results presented, the labels given by a non-expert are significantly different from those given by an expert, and thus the layman's labels are not reliable. However, the results also showed that a machine learning model fit to human-defined labels can effectively replace the human in labeling future flight observations. For this reason, we believe ML can be used to reduce manpower hours in labeling large quantities of flight data. We expect this conclusion is generalizable to other, unrelated domains.

[1] A significant difference is one in which Cohen's kappa coefficient is less than 0.90 (where 1 is the maximum possible value).

[2] For logistic regression, QDA, and the ridge, model prediction performance for the layman labels is similar to the prediction performance for the expert labels, but the performance is relatively bad in both cases.

References

[1] E. Y. Rodin and M. Amin, "Maneuver prediction in air combat via artificial neural networks," *Computers and Mathematics with Applications*, vol. 24, no. 3, pp. 95–112, 1992.

[2] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*. New York: Springer, 2013.

[3] C. M. Bishop, *Pattern recognition and machine learning*. New York: Springer, 2006.

[4] Z. Blanks, A. Sedgwick, B. Bone, and A. Mayerchak, "Identification of flight maneuvers and aircraft Types utilizing unsupervised learning with big data," in *Systems and Information Engineering Design Symposium*, (Charlottesville, VA), pp. 180–185, IEEE, 2017.

[5] J. Sim and C. C. Wright, "The Kappa Statistic in Reliability Studies: Use, Interpretation, and Sample Size Requirements," *Physical Therapy*, vol. 85, pp. 257–268, 03 2005.