Theoretically, for an attribute, lower mean information lead to higher IG, means that most instances having same class would be distributed into the same values of this attribute. So, for each value, unlike the data distribution in training data, would contain more homogeneous instances, data become more "predictable" after filtering with this attribute.

Naive Bayes use prior and posterior probabilities to predict the most likely class, where prior probability could be seen as "fixed" term for different attributes since it only describes the class distribution on the whole training set. However, posterior probability (P(a=v|c)) could be vary, for attribute with higher IG, since each value contains more homogeneous instances, P(c|a=v) would be higher for one class and lower for the others. Therefore, according to Bayes theory, posterior and final prediction probability would be higher for one specific class and lower for others, lead to more “confident” prediction to that specific class.

Therefore, usually NB classifier will perform better with many high IG attributes shown since they provide "powerful" evidence that could “divide” classes, give higher posterior probability to correct class, make prediction less ambiguous and bias. For instance, files anneal.csv, nursery.csv, mushroom.csv, high accuracy (> 90%) has been achieved with many high IG attributes (IG > 0.2), for some others like cmc.csv, breast-cancer.csv, low accuracy occurred (just like accuracy we could obtained from dummy classifier) with no high IG attributes.

There are exceptions like primary-tumor.csv and hypothyroid.csv, having high accuracy with no high IG attributes or low accuracy with high IG attributes present. For hypothyroid.csv, it is because classes in it are highly unequally distributed (3012 vs 151), choosing the majority is good enough to achieve high accuracy (95%). Also, most attributes only contain two values and not that informative, it is unlikely possible for them to distinguish minority from such large amount of majority thus result in all low IG. But since choosing majority is good enough, high accuracy is achieved even without informative attributes. For primary-tumor.csv, since IG prefers highly-branching attribute and this file contains so many missing values(based on my algorithm are all eliminated), also some attributes and classes only have few instances, so highly-branched attributes that could separate these small classes will get high IG, since it looks divides classes better. But they are not representative to large classes thus could not help predicting these big classes correctly, which finally lead to a low accuracy with the presence of high IG attributes.

Q4:

As shown in appendix 1, for most of files like anneal, hepatitis and car, precision, recall and f1 score on different average measurements all went down a little bit (0.1 - 0.6) after implementing 10-fold cross validation. It is actually a more realistic proxy for the accuracy since test on training data usually will cause overfitting problem. Test on training is useful in some cases as descriptive model when we just want to know how well the model describes the dataset. However generally we want a more generic model that could predict unseen instances. In test on training data, all the test data have been put into NB model and adjust both prior and posterior probabilities, change the prediction probability on desired class (input while training) higher. Since the model so "perfectly" that even fits outliers, noise and other variance, it is expected that the model will give higher accuracy output. Cross validation introduces a mechanism that still train and test the whole dataset eventually but not in the same run, it partition data to introduce new instances so that it could better reflect generalize case, so that probabilities inside NB model won't be adjusted by test cases to better predict them, the model we get is more realistic and the result of accuracy is more close to general accuracy. Therefore, cross validation helps reduce the overfitting problem thus results in a slightly decrease on accuracy measurements.

However, there two exceptions. First one is files like hypothyroid, nursery and mushroom that accuracy didn't decrease a lot (<0.001). It is because they are either having highly unbalanced class distribution (hypothyroid: 3012 vs151) or having attributes that are so informative (with IG > 0.9). These features remain almost the same in both training and testing data after partition, result in nearly the same probabilities measurement in NB models. The test instances don’t introduce new variance or unusual cases that violates these features (high IG attributes or unequal distribution). Therefore, the accuracy keeps almost the same after using cross validation. The other one is happened on primary-tumor that accuracy decreased a lot after implementing cross validation, one possible reason could be there are too many types of classes and attributes with just small amount of instances and many missing values, so that training data after partition is not "representative" enough for the whole data set. Also, the testing data introduces too many new instances with missing values that there are few or none evidences in the training data to classify them correctly, as prediction probabilities are all pretty small (including epsilon into it). Consequently the accuracy degrades a lot after implementing cross validation.