* <https://link.springer.com/content/pdf/10.1007/s10994-005-4258-6.pdf?pdf=button>
  + Still assumes independence, but a weaker form of independence
  + According to this author, assuming dependence in an algorithm causes it to have a large computational overhead, which is why they’re only assuming a bit of dependence
  + Naive Bayes calculated P(x|y)=P(x1 | y)\*P(x2 | y) \* …
  + The AODE variant used in the paper classifies by picking a random “i” and calculating P(x1 | xi, y)\*P(x2 |xi, y)\*...
  + They compare the prediction error, bias, variance, training time and classification time of AODE (Averaged One-Dependence Estimators) to those of Naive Bayes, ODE (One-Dependence Estimators), bagged ODE, LBR, TAN, and SP-TAN.
  + AODE accuracy still ended up being lower than J48 on dependent datasets, but it’s better than NB, LBR, TAN, and SP-TAN
* [Naives Bayes with Weight Function](https://dl.acm.org/doi/pdf/10.1145/1089815.1089826)
  + Designed for binary classifications
  + Uses a weight function
  + Cares about computational overhead and CPU time
  + Has a weight function to reduce the effects of outliers
  + Works with discrete and continuous variables
* [Hidden Naive Bayes](https://www.aaai.org/Papers/AAAI/2005/AAAI05-145.pdf)
  + Gives each node a hidden “parent node”, which is influenced by all other nodes (Uses same formula as Naive Bayes, but with “attribute X = y” as another precondition
  + Faster than CN-TAN (Less connections)
* Gaussian Naive Bayes
  + Seems like it only works for quantitative data? Class variable can be qualitative though (classification)